# United Aircraft Research Laboratories

UNITED AIRCRAFT CORPORATION

EAST HARTFORD, CONNECTICUT

Report E-910350-11

Low-Thrust Guidance Study

Contract NAS8-20119
Final Report

UNCLASSIFIED

REPORTED BY W. R. Fimple
W. R. Fimple

C. P. Van Dine
C. P. Van Dine

APPROVED BY R. A. LeSchack, for
J. L. Cooley

DATE July 1966

NO. OF PAGES 76

COPY NO. _____

Report E-910350-11

# Low-Thrust Guidance Study

Contract NAS8-20119
Final Report

## TABLE OF CONTENTS

TABLE OF CONTENTS
(contd.)

Report E-910350-11

# Low-Thrust Guidance Study

Contract NAS8-20119
Final Report

## SUMMARY

A computer algorithm has been developed for determining minimum-time optimal control for continuous low-thrust propulsion systems operating in an inverse-square gravity field. In the guidance procedure developed, instead of using a linearized solution about a precomputed nominal trajectory, the two-point boundary value variational problem is resolved at a number of updating points throughout the trip with the present observed state supplying the new initial boundary conditions. It is found that these updating points must be more frequent as the trip progresses. The algorithm employs the implicit finite-difference Newton-Raphson algorithm to descretize and solve the variational differential equations with the two-point boundary value problem as a sequence of linear finite-difference equations. Although the algorithm requires a substantial fraction of the core storage capacity to an IBM-7094 computer, it may be possible that it could be adapted to an advanced on-board computer.

The algorithm has been applied in a guidance study involving a nominal 266-day minimum-time Earth-Mars trajectory where the planets are considered to be massless points. The most important finding of this study is the high sensitivity of the control to small guidance errors in the final phase of the trip.

Applications of the algorithm to low-thrust planetocentric flight have been limited to establishing nominal trajectories and associated control programs. A discussion of the same problem in the time-varying field generated by the sun and planets considered as point masses is also presented, including the variational formulation and an adaptation of the algorithm to accommodate variable mesh-point spacing. Finally, some recommendations for further research in this area are given.

## INTRODUCTION

The practical problem of guidance of a space vehicle is concerned with two questions: what is the present state (i.e., position and velocity); and how should the spacecraft be controlled from the present state in order to arrive at a given destination. Although the first question, the problem of navigation, is by no means trivial or of minor importance, the present study deals exclusively with the second.

The problem treated in the present study is the determination of optimum control for a constant specific-impulse, power-limited space vehicle to accomplish rendezvous in an inverse-square central force field in minimum time. Requiring continuous thrusting, the minimum-time operation of a low-thrust vehicle is the limiting case of the constant-thrust, optimum (single or multiple) coast mode of operation which maximizes payload for a fixed trip time. From a guidance point of view, the two powered phases of constant thrust with coast trajectories are treated as minimum-time problems (although the first phase has additional complexity). This minimum-time problem has been successfully solved for two regimes: (1) for a thrust acceleration of the order of one-sixth the local gravity acceleration, corresponding to interplanetary flight; and (2) for a local thrust-to-weight ratio of the order of one-thousandth, corresponding to planetocentric escape and capture.

Unlike guidance schemes where the strategy is always given with respect to a precomputed nominal trajectory, the current approach to the problem is to resolve the complete variational system using the current observed state for new boundary conditions. Referred to as updating-the-trajectory, this procedure is repeated many times during the trip. Described in Ref. 1 and further developed in this report, the basic numerical method employed in this guidance program is the implicit finite-difference Newton-Raphson algorithm. Briefly, this algorithm substitutes a set of linear finite-difference equations for the system of differential equations (Euler-Lagrange equations and equations of motion) which is solved (including boundary conditions) by applying a generalization of the classical Newton iteration to an approximate solution.

The method of second-variation guidance produces a new control, at each application, which is a linearized solution about the precomputed nominal. It is interesting to note that this corresponds roughly to a single iteration of the Newton-Raphson scheme, i.e., each iteration produces the control which would be the solution if the problem were linear. However, multiple iterations can achieve solutions which are strikingly non-linear with respect to the nominal as is shown in this report.

At this point, a few comments to qualitatively compare the problems of low-thrust guidance and control with those for high thrust may be useful. In one respect, the low-thrust guidance and control problems are simpler and less

demanding on hardware. Whereas the propulsion time (and control time) of a high-thrust space vehicle is a very small fraction of the total trip time, a low-thrust vehicle usually has control over a major portion of the total trip. Hence, while very small control errors are magnified into huge miss distances in the high-thrust case, fairly large errors in control can be tolerated in low-thrust space flight at least throughout the first half of the trip. On the other hand, the low-thrust vehicle is severely limited in the magnitude of the control that can be applied in any given small time period. For this reason, the terminal guidance phase is an extremely critical period.

Finally, it should be emphasized that the scope of the study does not include a computer simulation of low-thrust space flight; that is, no attempt has been made to determine expected guidance errors in any statistical sense, but rather sequences of possible errors have been assumed at a number of updating points along the trajectories in the sample problems.

## VARIATIONAL EQUATIONS

In first-order form the equations of motion of a point mass subject to constant thrust in an inverse-square gravity field are:

$$\dot{x}_i = u_i \qquad i = 1, 2, 3 \tag{3.1}$$

$$\dot{u}_i = \frac{a_o \cos \alpha_i}{1 + bt} - \frac{x_i}{r^3} \qquad i = 1, 2, 3 \tag{3.2}$$

with the constraint on the direction cosines of the thrust vector

$$\sum_i \cos^2 \alpha_i - 1 = 0 \tag{3.3}$$

Here the $x_i$ are cartesian coordinates, r is the radius, t is time, $a_o$ is the initial thrust acceleration, and b is the fraction of initial mass expended per unit time. Cos $\alpha_i$ are the direction cosines of the thrust-acceleration vector. It is desired to determine the control $\alpha_i(t)$ which will minimize the time between two given sets of boundary and transversality conditions. Such an extremal must satisfy the Euler-Lagrange equations of the calculus of variations as well as the constraining equations 1, 2, and 3 above. The Euler-Lagrange equations can be determined from the variational Hamiltonian,

$$H = \sum_i \left[ \pi_i u_i + \lambda_i \left( \frac{a_o \cos \alpha_i}{1 + bt} - \frac{x_i}{r^3} \right) + \lambda_F \cos^2 x_i \right] \tag{3.4}$$

where $\pi_i$, $\lambda_i$, and $\lambda_F$ are Lagrange multipliers. The Hamiltonian must be locally stationary with respect to the control. Therefore,

$$\frac{\partial H}{\partial \alpha_i} = 0 = - \lambda_i \frac{a_o \sin \alpha_i}{1 + bt} - 2\lambda_F \sin \alpha_i \cos \alpha_i \tag{3.5}$$

and

$$\lambda_F = \frac{a_o p}{2(1 + bt)} \tag{3.6}$$

Substituting (3.6) into (3.5) gives the control variables in terms of the adjoint variables.

$$\cos \alpha_i = \frac{\lambda_i}{p} \tag{3.7}$$

where $p^2 \equiv \sum_i \lambda_i^2$

The Euler-Lagrange equations are now determined from Hamilton's canonical equations:

$$\dot{\lambda}_i = - \frac{\partial H}{\partial u_i} ; \; u_i = \frac{\partial H}{\partial \lambda_i} \tag{3.8}$$

$$\dot{\pi}_i = - \frac{\partial H}{\partial x_i} ; \; x_i = \frac{\partial H}{\partial \pi_i} \tag{3.9}$$

$$\dot{\lambda}_i = - \pi_i \tag{3.10}$$

$$\ddot{\lambda}_i = - \dot{\pi}_i = \frac{\partial H}{\partial x_i} \tag{3.11}$$

The second two relations of equations (3.8) and (3.9) give the equations of motion. Operating on the Hamiltonian as indicated in Equation (3.11) gives

$$\ddot{\lambda}_i + \left( \lambda_i - \frac{3 x_i s}{r^2} \right) r^{-3} = 0 \qquad i = 1, 2, 3 \tag{3.12}$$

where

$$s = \sum_{i=1}^{3} x_i \lambda_i$$

which are the Euler-Lagrange equations written in second-order canonical form. The adjoint variables, $\lambda_i$, are the Cartesian components of the primer vector, Ref. 2. The equations of motion written in the same form with the control expressed in terms of the adjoint variables are

$$\ddot{x}_i - \frac{a \lambda_i}{p} \frac{x_i}{r^3} = 0 \qquad i = 1, 2, 3 \tag{3.13}$$

where $a = a_o/(1 + bt)$

Equations (3.12) and (3.13) are necessary conditions for an extremal arc.

## Transversality Condition for Minimum-Time Rendezvous

The minimum-time problem can be formulated naturally as a Mayer-type problem in the calculus of variations where it is desired to minimize a function $\varphi$ of the final boundary conditions. For the particular case at hand, the function $\varphi$ is merely the final time itself (or equivalently $\varphi = Ct$ where C is an arbitrary constant).

The general transversality condition at the final boundary is

$$d\varphi + \sum_i \left[ - \lambda_i \frac{a_o \cos \alpha_i}{1 + bt} + \frac{\lambda_i x_i}{r^3} \right] dt + \sum_i \lambda_i du_i = 0 \qquad (3.14)$$

where the $u_i$ are the rectangular components of velocity. The position and velocity of the body with which rendezvous is to take place can be described parametrically in time.

$$x_i = f_i(t) \qquad (3.15)$$

$$\dot{x}_i = u_i = \dot{f}_i(t) \qquad (3.16)$$

The position and velocity components of the vehicle and the target body must become identical at final time. Therefore, differentiating and substituting equation (3.16) into equation (3.14) gives the final form of the transversality condition.

$$\left[ C - \frac{a_o p}{1 + bt} + \sum_i \lambda_i \left( \frac{x_i}{r^3} + \dot{f}_i \right) \right] dt = 0 \qquad (3.17)$$

where the relations $\cos \alpha_i = \frac{\lambda_i}{p}$ and $d\varphi = Cdt$ have been used.

If the rendezvous is taking place with a free-falling body (such as a planet) each term of the summation in equation (3.17) is identically zero. Since equation (3.17) must hold for any small variation of final time, the coefficient of dt must vanish. This shows that the magnitude of the primer vector can be arbitrarily scaled since C is an arbitrary constant. This result stems from the fact that minimizing final time is identical to minimizing any constant multiple of final time.

## Equations for the Numerical Solution

The numerical solution of this system of differential equations with two-point boundary conditions is achieved through the application of the finite-difference Newton-Raphson algorithm. The detailed construction of this algorithm for general systems is discussed in the next section. At this point the actual algebraic expressions used to solve the equations derived above are given.

Equations (3.13) and (3.12) form the basic system of six second-order differential equations. The six conditions at the initial boundary are the instantaneous position and velocity of a spacecraft, and the six conditions at the final boundary are the time-varying position and velocity of a massless target point moving in a three-dimensional Keplerian ellipse. The final equation is the arbitrary scaling of the primer vector

$$p(0) = k \tag{3.18}$$

Define $\ddot{x}_i = f_i$ (from 3.13) and $\ddot{\lambda}_i = g_i$ (from 3.12). Because of the symmetry of the $f_i$ and $g_i$ the Jacobian of these equations can be written in terms of the following eight expressions.

$$\frac{\partial f_i}{\partial x_i} = \left( \frac{3x_i^2}{r^2} - 1 \right),$$

$$\frac{\partial f_i}{\partial x_\nu} = \frac{3x_i x_\nu}{r^5}, \qquad i \neq \nu,$$

$$\frac{\partial f_i}{\partial \lambda_i} = \frac{a}{p^3}\left(p^2 - \lambda_i^2\right), \tag{3.19}$$

$$\frac{\partial f_i}{\partial \lambda_\nu} = -\frac{a}{p^3}\lambda_i \lambda_\nu, \quad i \neq \nu,$$

$$\frac{\partial g_i}{\partial x_\nu} = \frac{3}{r^5}\left[2x_i\lambda_i - 5\left(\frac{5x_i}{r^2} - 1\right)\right]$$

$$\frac{\partial g_i}{\partial x_\nu} = \frac{3}{r^5}\left(\lambda_i x_\nu + \lambda_\nu x_i - \frac{5x_i x_\nu s}{r^2}\right), \quad i \neq \nu$$

$$\frac{\partial g_i}{\partial \lambda_i} = \frac{\partial f_i}{\partial x_i}, \quad \text{and} \quad \frac{\partial g_i}{\partial \lambda_\nu} = \frac{\partial f_i}{\partial x_\nu}, \quad i \neq \nu,$$

The form of this Jacobian allows a significant saving in the computational storage requirement as is pointed out in the appendix of Ref. 1.

Equation (3.18) can be written as

$$\sum_i \lambda_i^2(0) = k^2, \tag{3.20}$$

and the partial derivative with respect to each $\lambda_i(0)$ is trivial. In addition, the partial derivative

$$\frac{\partial f_i}{\partial t} = -\frac{a\,b}{(1+bt)}\,\frac{\lambda_i}{p} \tag{3.21}$$

is also required. The only remaining derivatives with respect to time which need to be considered at this point are those of the state of the target which are well known.

## NUMERICAL SOLUTION OF THE NONLINEAR BOUNDARY VALUE PROBLEMS

This section gives the formal construction of the implicit, finite-difference, Newton-Raphson algorithm for the numerical solution of systems of ordinary differential equations with split boundary conditions. The generality of systems considered will include all types encountered in this report. This construction parallels Ref. 1 up to the treatment of transversality conditions. The approach taken is to begin with a basic statement of the problem and proceed to each extension of the method individually, instead of immediately placing the entire problem in its most general formulation.

### The Basic Problem

Consider, therefore, the system of equations,

$$\frac{d^2y_i}{dx^2} - f_i(y_1, \ldots, y_m, x) = 0, \quad 1 \leq i \leq m , \tag{4.1}$$

defined on the interval,

$$a \leq x \leq b ,$$

with boundary conditions

$$y_i(a) = \alpha_i \quad \text{and} \quad y_i(b) = \beta_i .$$

The $f_i$ are, in general, nonlinear expressions, the only restriction on them being that all their partial derivatives with respect to the $y_i$ must exist. A solution of this system is a vector $\mathbf{y} = (y_1, \ldots, y_m)$ of functions of x which satisfy the equations and boundary conditions on the given interval. (For conciseness, vectors will be displayed as rows but should be interpreted for computational purposes as column vectors.)

Clearly one can evaluate equations (4.1) for any vector of the proper form, and if $\mathcal{P}$ is taken to be the nonlinear operator defined by the system, the problem can be stated as finding a root of the form $\mathcal{P}(\mathbf{y}) = 0$. The Newton-Raphson iteration for finding roots of nonlinear, algebraic equations can be generalized to handle such operator equations (Ref. 3). That is, given an appropriate initial guess, or starting solution, $\mathbf{y}_0$, the iteration

$$\mathcal{Y}_{k+1} = \mathcal{Y}_k - \left[\mathcal{P}'(\mathcal{Y}_k)\right]^{-1} \mathcal{P}(\mathcal{Y}_k) , \tag{4.2}$$

where $\mathcal{P}'$ is in some sense which we need not define here the "derivative" of $\mathcal{P}$, will usually yield a sequence $\{\mathcal{Y}_k\}$ which converges to the desired solution $\mathcal{Y}$. The goal here is to write the iteration (4.2) for the system (4.1) in a manner which is numerically tractable. The notation can be simplified by defining $\mathcal{D}_k = \mathcal{Y}_{k+1} - \mathcal{Y}_k$ and rewriting (4.2) without the iterative subscript k as

$$-\left[\mathcal{P}'(\mathcal{Y})\right]\mathcal{D} = \mathcal{P}(\mathcal{Y}) . \tag{4.3}$$

It is understood that $\mathcal{D}$ is the change in $\mathcal{Y}$ at the given step in the iteration. Equation (4.3) will ultimately be written as a large, finite, matrix equation where $-[\mathcal{P}'(\mathcal{Y})]$ becomes a matrix, $\mathcal{P}(\mathcal{Y})$ the known right hand vector, and $\mathcal{D}$ the unknown vector to be solved for.

To proceed, one now represents each function $y_i$ by its value at n points $x_j$, and $y_i(x_j)$ will most often be denoted by $y_{ij}$. The following notation will now be useful. $Y_j = (y_{ij}, \ldots, y_{mj})$, and in this representation $\mathcal{Y} = (Y_1, \ldots, Y_n)$. Thus $\mathcal{Y}$ has been reduced to an ordered nxm tuple of numbers. (Note that it is a vector, not a matrix.) Similarly, $\mathcal{D} = (D_1, \ldots, D_n)$ where $D_j$ is the vector of iterative charges at $x_j$, and using the vector function $F = (f_1, \ldots, f_m)$, it will become practical to write $\mathcal{P} = (P_1, \ldots, P_n)$. $F_j$ will be used to denote $F(Y_j, x_j)$.

Initially, the points $x_j$ will be equally spaced according to the formula:

$$x_j = a + (j-1)h , \tag{4.4}$$

where

$$h = (b-a)/(n-1) . \tag{4.5}$$

Note that $x_1 = a$, and $x_n = b$. However, a method will also be discussed whereby the $x_j$ may be variably spaced according to some appropriate scheme.

Returning now to equation (4.1), the standard three-point formula,

$$\Delta_2(y_{ij}) = \frac{y_{ij-1} - 2y_{ij} + y_{ij+1}}{h^2} , \tag{4.6}$$

is used to approximate the second derivatives, $\frac{d^2y_i}{dx^2}$. Combining (4.1) and

(4.6) and using some elementary algebra, one can now write the operator $\mathcal{P}$ for the given, discrete representation of $\mathcal{Y}$.

$$
\begin{aligned}
P_1 &= (\alpha_1, \ldots, \alpha_m) - Y_1, \\
P_j &= Y_{j-1} - 2Y_j + Y_{j+1} - h^2 F_j, \qquad 2 \leq j \leq n-1, \\
P_n &= (\beta_1, \ldots, \beta_m) - Y_n .
\end{aligned}
\tag{4.7}
$$

Now the iteration (4.3) can be immediately put in finite matrix form.

$$
\begin{bmatrix}
I & & & & & & \\
-I, & 2I + h^2 F_2', & -I & & & & \\
 & \ddots & \ddots & \ddots & & & \\
 & & -I, & 2I + h^2 F_{j-1}', & -I & & \\
 & & & -I, & 2I + h^2 F_j', & -I & \\
 & & & & \ddots & \ddots & \ddots \\
 & & & & & -I, & 2I + h^2 F_{n-1}', \; -I \\
 & & & & & & I
\end{bmatrix}
\begin{bmatrix} D_1 \\ D_2 \\ \vdots \\ D_{j-1} \\ D_j \\ D_{j+1} \\ \vdots \\ D_{n-1} \\ D_n \end{bmatrix}
=
\begin{bmatrix} P_1 \\ P_2 \\ \vdots \\ P_{j-1} \\ P_j \\ P_{j+1} \\ \vdots \\ P_{n-1} \\ P_n \end{bmatrix}
\tag{4.8}
$$

I is the nxm identity matrix, and $F_j'$ are the Jacobian matrices

$$\begin{pmatrix} \frac{\partial f_1}{\partial y_1}, \ldots, \frac{\partial f_1}{\partial y_m} \\ \vdots \\ \frac{\partial f_m}{\partial y_1}, \ldots, \frac{\partial f_m}{\partial y_m} \end{pmatrix} \tag{4.9}$$

evaluated using $Y_j$ and $x_j$.

The matrix in equation (4.8), which can be denoted as $-[\mathcal{P}'(a_j)]$, is a special case of the general block tri-diagonal matrix

$$\begin{bmatrix} B_1, C_1 & & & \\ A_2, B_2, C_2 & \bigcirc & & \\ \ddots & \ddots & \ddots & \\ \bigcirc & A_{n-1}, B_{n-1}, C_{n-1} & & \\ & A_n, B_n & & \end{bmatrix} \tag{4.10}$$

whose solution may be easily obtained by using the direct elimination formulas

$$W_1 = B_1^{-1}C_1; \quad W_j = (B_j - A_jW_{j-1})^{-1}C_j, \qquad 2 \leq j \leq n, \tag{4.11}$$

$$G_1 = B_1^{-1}P_1; \quad G_j = (B_j - A_jW_{j-1})^{-1}(P_j - A_jG_{j-1}), \qquad 2 \leq j \leq n,$$

followed by the back substitution

$$D_n = G_n; \quad D_j = G_j - W_j D_{j+1}, \qquad n - 1 \geq j \geq 1 , \tag{4.12}$$

for the solution. For systems arising from variational problems - $[\mathcal{P}'(y)]$ is known to be positive-definite in the vicinity of a solution of interest (i.e., one not containing a conjugate point). This condition guarantees that the inverses indicated in (4.11) exist, and that the system is well conditioned. Notice, also, that the initial and final $D_j$ may be eliminated from the system prior to the application of formulas (4.11) and (4.12). Here, and in the subsequent discussion of transversality conditions, such a move is completely appropriate.

Once a starting solution has been determined, the iteration (4.8) is repeated until the element with largest absolute value in $\mathcal{D}$ has been brought below some reasonable epsilon. Due to the quadratic convergence of the Newton-Raphson algorithm, this convergence criterion can usually be achieved in fewer than ten iterations.

## General Boundary Conditions

The simple boundary conditions considered for (4.1) are clearly not adequate for the majority of problems. The next step in the construction of the algorithm is to show how more general conditions can be handled. Transversality conditions can be included at either boundary in a similar and symmetric manner, and the analysis presented below in terms of the initial boundary is easily applied to the final boundary.

Consider, therefore, the set of conditions

$$\varphi_i(y_1(a), \ldots, y_m(a), y_1'(a), \ldots, y_m'(a), a) = 0, \quad 1 \leq i \leq m. \tag{4.13}$$

The formula,

$$\Delta_1(y_{ij}) = \frac{- y_{ij-1} + y_{ij+1}}{2h} \tag{4.14}$$

is used to evaluate these first derivatives to the same order of accuracy that (4.6) brings to the second derivatives. However, applying (4.14) at the initial point a assumes knowledge of the functions $y_i$ at the point $x_0 = a - h$, and it is necessary to introduce the quantities $Y_0 = (y_1(a - h), \ldots, y_m(a - h))$ and the associated $D_0$ and $P_0$. Now (4.13) can be written, in a manner analogous to that used above, as

$$\Phi(Y_1, Y_0, Y_2, x_1) = 0 , \tag{4.15}$$

and (4.8), the discrete form of $\mathcal{P}$, becomes

$$\begin{aligned} P_0 &= \Phi , \\ P_j &= Y_{j-1} - 2Y_j + Y_{j+1} - h^2 F_j , \qquad 1 \le j \le n-1 , \\ P_n &= (\beta_1, \ldots, \beta_n) - Y_n . \end{aligned} \tag{4.16}$$

In order to have a sufficient number of equations to solve for the new unknowns, $D_0$, the discrete form the differential equations (4.1) must be also written at the boundary, a. The matrix equation (4.8) is only affected in the upper rows; which become

$$\begin{bmatrix} \Phi_0', \Phi_1', \Phi_2' & & \\ -I, 2I + h^2 F_1', -I & & \\ -I, 2I + h^2 F_2', -I & & \\ \ddots & \ddots & \ddots \end{bmatrix} \begin{bmatrix} D_0 \\ D_1 \\ D_2 \\ \vdots \end{bmatrix} = \begin{bmatrix} P_0 \\ P_1 \\ P_2 \\ \vdots \end{bmatrix} \tag{4.17}$$

where the $\Phi_j'$ are the Jacobian matrices $\{\frac{\partial \varphi_i}{\partial y_v}\}$ , $1 < i, v < m$, evaluated using $Y_j$ and $x_j$. The matrix in (4.17) can be reduced to block tri-diagonal form either by eliminating $D_0$ or by considering the first two rows of subblocks as one. Thus, the solution equations (4.11) and (4.12) still apply, and the iteration can be carried out as above.

## Additional Parameters

It is not unusual for a system of equations, especially those arising from a variation problem, to depend not only on a vector of unknown functions, $y$ , but also on a vector of unknown parameters. Next, the case of including one such unknown parameter into the construction of the algorithm will be treated, and the extension to any number of such parameters can be made a similar fashion.

Consider, therefore, in lieu of (4.1), the system of equations

$$\frac{d^2y_i}{dx^2} - f_i(y_1, \ldots, y_m, r, x), \qquad 1 \leq i, \leq m, \tag{4.18}$$

where r is an unknown scaler parameter. It is immediately obvious that an additional constraint,

$$g(\mathcal{Y}, r) = 0 , \tag{4.19}$$

is needed to define the system. What may not be so obvious is that g may not depend on r at all. That is, r may enter the system completely through the $f_i$, and g only serves to complete the definition of $\mathcal{Y}$.

The analysis proceeds exactly as before except that the Jacobian matrices (4.9) are now augmented by an additional column,

$$F' = \begin{pmatrix} \frac{\partial f_1}{\partial y_1}, \ldots, \frac{\partial f_1}{\partial y_m}, \frac{\partial f_1}{\partial r} \\ \cdot \quad \cdot \quad \cdot \\ \cdot \quad \cdot \quad \cdot \\ \cdot \quad \cdot \quad \cdot \\ \frac{\partial f_m}{\partial y_1}, \ldots, \frac{\partial f_m}{\partial y_m}, \frac{\partial f_m}{\partial r} \end{pmatrix} , \tag{4.20}$$

which necessitates changing the form of the matrix equation (4.8).

The following notation will be used to describe these changes. Define $R_j = (h^2\frac{\partial f_1}{\partial r}, \ldots, h^2\frac{\partial f_m}{\partial r})$, $2 \leq j \leq n - 1$, $R_1$ and $R_n$ are the vectors of partial derivatives with respect to the boundary condition equations, and $\mathcal{R} = (R_1, \ldots, R_n)$. Define $G_j = (\frac{\partial g}{\partial y_1}, \ldots, \frac{\partial g}{\partial y_m})$, and $\mathcal{G} = (G_1, \ldots, G_n)$. (Note that $G_j$ and $\mathcal{G}$ are row vectors.) Also, define $d_r$ as the iterative change in r at the k th iteration.

Thus, the matrix iteration including $d_r$ can be written as an augmented version of (4.8).

$$\begin{bmatrix} -[\mathcal{P}'(\mathcal{Y})] & \begin{matrix} R_1 \\ R_2 \\ \vdots \\ R_n \end{matrix} \\ G_1, G_2, \ldots, G_n, \frac{\partial g}{\partial r} & \end{bmatrix} \begin{bmatrix} D_1 \\ D_2 \\ \vdots \\ D_n \\ d_r \end{bmatrix} = \begin{bmatrix} P_1 \\ P_2 \\ \vdots \\ P_n \\ -g \end{bmatrix} . \tag{4.21}$$

Equation (4.21) is solved by first using the previously defined method for the matrix $-[\mathcal{P}'(\mathcal{Y})]$ with multiple right hand sides. That is, $\mathcal{U}$ and $\mathcal{V}$ are defined by the equations

$$-[\mathcal{P}'(\mathcal{Y})]\mathcal{U} = \mathcal{R} \quad \text{and} \quad -[\mathcal{P}'(\mathcal{Y})]\mathcal{V} = \mathcal{P} . \tag{4.22}$$

which lead to, by elementary matrix algebra, the relations

$$d_r = (\mathcal{G}\cdot\mathcal{U})^{-1}(\mathcal{G}\cdot\mathcal{V} - g) ,$$

and (4.23)

$$\mathcal{D} = \mathcal{V} - \mathcal{U}d_r .$$

Equations (4.23) become more complicated if $d_r$ is a vector, i.e., if there is more than one unknown parameter, but in their present form the generalization to that case is immediate.

Since most of the calculations in equations (4.11) and (4.12) involve the matrix, it should be noted that there is a minimum of extra work required to compute the two vectors, $\mathcal{U}$ and $\mathcal{V}$. Also, in most cases, the vector $\mathcal{G}$ is quite sparse, a fact which considerably simplifies the calculation of the inner products in (4.23).

## Variable End Points

A particular problem of interest, and one which requires further insight, is the case where one of the end points, say b, is itself an unknown parameter. Any minimum time problem is such a variable end-point problem. Going back to

equations (4.4) and (4.5), one sees that each value of the independent variable, $x_j$, and the mesh spacing, h, depend on b. Thus, if b is unknown, the only truly independent variable remaining is the mesh point number j. However, once these facts are realized, the analysis presented above is completely applicable. The point of the matter is that every occurrence of $x_j$ and h must be considered as a function of b when the elements of the vector $\mathbf{Q}$ are formed. This situation gives rise to some complicated algebraic expressions, but no real problems.

## Variable Mesh Spacing

The final consideration given here in the construction of the algorithm is the inclusion of variable mesh spacing. The attempt here is to formulate the variable mesh spacing in a manner which allows all of the above analysis to remain applicable. The spacing of the points where the functions $y_i$ are evaluated is given by an arbitrary sequence $\{\tilde{x}_j\}$. The relative spacing of the elements of this sequence is preserved in the spacing of the actual $x_j$ by the formula

$$x_j = a + \tilde{x}_j(b/\tilde{x}_n) \ , \tag{4.24}$$

which now replaces (4.4). Similarly, one can define $h_j = x_{j+1} - x_j$, $1 \leq j \leq n-1$, and write

$$h_j = (\tilde{x}_{j+1} - \tilde{x}_j)\ (b/\tilde{x}_n) \tag{4.25}$$

in place of (4.5). Equations (4.6) and (4.19) are no longer valid and must be replaced by the appropriate divided difference formulas,

$$\Delta_2(y_{ij}) = \frac{2}{h_{j-1}h_j(h_{j-1} + h_j)} (h_j y_{ij-1} - (h_{j-1} + h_j)y_{ij} + h_{j-1}y_{ij+1}) \ ,$$

and (4.26)

$$\Delta_1(y_{ij}) = \frac{1}{h_{j-1}h_j(h_{j-1} + h_j)} (-h_j^2 y_{ij-1} + (h_j^2 - h_{j-1}^2)y_{ij} + h_{j-1}^2 y_{ij+1}) \ .$$

It appears that the algebraic complications, especially those associated with the variable end-point problem, have been increased; however, a little manipulation can facilitate matters. Defining $u = h_j/h_{j-1}$, $v = (h_{j-1} + h_j)/h_{j-1}$, and $w = h_j(h_{j-1} + h_j)/2$, only w is a function of b, and

$$\Delta_2(y_{ij}) = \frac{uy_{ij-1} - vy_{ij} + y_{ij+1}}{w} . \qquad (4.27)$$

Thus, the only change in equation (4.7) is

$$P_j = uY_{j-1} - vY_j + Y_{j+1} - wF_j , \qquad 2 \leq j \leq n - 1 , \qquad (4.28)$$

and the general row of (4.8) becomes

$$- uD_{j-1} + (v + wF_j')D_j - D_{j+1} = P_j . \qquad (4.29)$$

Therefore, the general, block tri-diagonal, form of the solution remains the same when variable mesh spacing is introduced in the above manner.

## GUIDANCE FOR LOW-THRUST INTERPLANETARY FLIGHT

The implicit finite-difference Newton-Raphson algorithm has been applied to a guidance study involving a nominal 266-day minimum-time Earth-Mars trajectory. The flight is assumed to take place in the solar central force field and the two planets are assumed to be massless points. Initially the optimum nominal trajectory and associated control were determined. Then a sequence of arbitrary but physically possible guidance errors were assumed at four points along the nominal trajectory corresponding roughly to 1/5, 2/5, 3/5, and 4/5 of the total trip time. The resulting changes were obtained and are presented herein.

The purpose of the guidance study is more to exercise and demonstrate the capabilities of the algorithm rather than to explore the practical problems of low-thrust interplanetary guidance in detail. However, the results of the limited cases investigated are discussed in detail in order to establish a credibility for these results.

In the algorithm the updating process takes place as follows: Suppose the optimum trajectory and associated adjoint variables (i.e., the control variables) from certain boundary conditions at time $t_1$ exist in the computer and it is desired to update the trajectory to new boundary conditions at a later time $t_2$. (These new boundary conditions at $t_2$ will generally be the state on the old nominal from $t_1$ at time $t_2$ plus an error vector due to the fact that the nominal has not actually been followed as specified.) The section of the old nominal from $t_2$ to the final time, T, is used as a starting approximation for the updated trajectory from the new boundary conditions at $t_2$. In this updating process the number of mesh points is conserved. Therefore, it is necessary to interpolate between mesh points on the old nominal since there are only $(1 - t_2/T)$ N mesh points in the segment between $t_2$ and T (where N is the total number of mesh points).

It may be that the required new boundary conditions at $t_2$ are sufficiently remote from the nominal at this point that the Newton-Raphson iteration cannot converge directly on the new solution. In this case, it is necessary to repeat the updating process a number of times for a sequence of smaller error vectors leading up to the required total error vector. It has been the authors' experience that convergence can always be obtained if small enough steps are taken, providing, of course, that the updated trajectory is of the same class as the old nominal; e.g., it makes the same number of circuits around the sun.

The density of updating points is somewhat arbitrary as far as the algorithm is concerned. It is obvious that the longer the time interval between updating points the larger will be the guidance errors involved. One limit would be practically continuous updating so that only the initial

control on each successive updated program actually would be followed. Operationally, taking into account factors related to onboard computational capabilities, accuracy of navigational equipment, and precision of thrusting control, there probably exists a best spacing of updating points along the trip. The results of this study show that the spacing certainly must get finer as the trip progresses.

## Numerical Accuracy Determination

Before proceeding to the results of the guidance study, a few preliminary results are presented to determine the numerical accuracy of the algorithm for the particular problem at hand. Figure 1 shows the resulting minimum total time for the nominal Earth-Mars trajectory as a function of the number of mesh points employed in the calculation. The scale on the right of the figure is in days while that on the left is in $\tau$ units which are the natural nondimensional units of time used in the calculations and result from setting the gravitational constant of the sun and the Earth's mean orbital radius each equal to unity ($1\tau = (2\pi)^{-1}$ years). Both scales are greatly expanded.

The results of Fig. 1 indicate that about 200 mesh points are adequate for this particular trajectory. The increase of trip time with increasing number of mesh points beyond 200 is very small. Figure 2 shows the same data plotted against mesh point spacing instead of number of mesh points. This plot can be interpreted more generally than the previous one because the number of mesh points is very closely connected with the particular problem at hand, while the mesh point spacing is not. It can be said generally that the mesh point spacing should be kept below about $2.5 \times 10^{-2}$ $\tau$ units.

Figure 2 indicates the sensitivity of the initial control to the number of mesh points employed in the nominal 266-day Earth-Mars trajectory. Theta ($\theta$) is the angle between the projection of the thrust vector unto the ecliptic plane and the direction of the vernal equinox. It will be referred to in the subsequent text and figures as the in-plane steering angle or control angle. Phi ($\phi$) is the out-of-plane angle between the thrust vector and the ecliptic plane. In the figure, both curves approach their limiting asymptotes at about 300 mesh points.

For any given problem, it is important to strike a good compromise between machine time and accuracy. The machine time is directly proportional to the number of mesh points. A good rule of thumb to follow is that an IBM 7094 computer can perform about 60 mesh-point iterations per second. Based on the results of Figs. 1 and 3, 200 mesh points were used throughout the Earth-Mars guidance study. It should be recalled that with each updating of the trajectory, the accuracy of the algorithm improves since the same number of mesh points are being put into a smaller time interval.

## The Nominal Trajectory and Control

Figure 4.0 is an ecliptic plane projection of the nominal Earth-Mars minimum-time constant-thrust trajectory and the associated thrusting program. The vectors in the figure represent the instantaneous acceleration on the vehicle due to the constant thrust. They are drawn to a scale of 0.5 in. = 0.001 meter/sec$^2$ and the directions of the vectors correspond to the in-plane steering angle, $\theta$. In fact, the vectors shown in the figure are the ecliptic plane projections of the computed three-dimensional thrust-acceleration vectors. Because the trajectory is almost coplanar (it starts in the ecliptic plane at Earth's orbit and comes up out of the paper until it has an altitude of about four thousandths of an AU at Mars) the ecliptic plane projection corresponds very closely to the actual trajectory and thrust-acceleration vectors.

The different apparent lengths of the thrust-acceleration vectors in Fig. 4.0 and the following figures are due to two causes: (1) When the vector is appreciably out of the ecliptic plane, it appears to be shorter because of foreshortening onto the ecliptic plane; (2) The thrust-acceleration vector increases monotonically with time because the vehicle has constant thrust and is losing mass.

It was assumed that the initial thrust acceleration of the vehicle is $5.337 \times 10^{-4}$ meters/sec$^2$ and that the vehicle exhausts $1.090 \times 10^{-5}$ fraction of its initial mass per kilosecond. These constants correspond to a specific impulse of 5000 sec, a powerplant specific weight of about 19 kg/kw, and a powerplant fraction of $\frac{1}{4}$.

The thrust-acceleration vectors are evenly spaced in time (every 13.3 days) along the trajectory and the marks on the orbits of Earth and Mars show the positions of these planets at the respective times for which the vectors are drawn. The date of departure from Earth is given in the figure as 2444180 which is the Julian date corresponding to November 25, 1979.

The nominal trajectory of Fig. 4.0 is divided into three thrusting phases. During the first phase from launch to about 120 days, the thrust acceleration vector is pointed generally outward away from the Earth thereby increasing the radial velocity. The second phase is a transition region between the first and third phases. Beginning at about 120 days, the thrust acceleration vector swings rapidly around clockwise until at 173 days it is pointed almost exactly along the vehicle - Mars line of sight, directly away from Mars. This situation persists during the entire third and terminal phase of thrusting from 173 days to the end of the trip.

It is interesting to note that the relative velocity vector of the vehicle with respect to Mars is practically along the line of sight between the two points throughout the entire terminal phase of the trip. This

orientation is not readily apparent from the figure, but it has been verified by the numerical data. In the terminal region the respective positions of the vehicle and Mars with respect to the sun are approximately the same so that the nonuniformity of the gravity field does not produce much effect and the situation could be approximated as occurring in a constant gravity field or field-free space. To simplify the visualization still further, assume that the thrust acceleration is constant. Then the guidance strategy in the terminal phase is simply to negate the relative velocity v in the time to go, T-t. For a given value of thrust acceleration, a, the velocity at the beginning of the terminal phase must satisfy the relation $v = a(T-t)$.

The control strategy for the real case in Fig. 4.0 is qualitatively the same, that is,to maneuver the vehicle onto a collision course with Mars with a relative velocity between the two when entering the terminal phase that roughly satisfies the $v = a(T-t)$ relationship.

The field-free analogue is very useful for a qualitative understanding of the central force field case. Thus the whole trajectory and control of Fig. 4.0 have the same qualitative characteristics as a minimum-time trajectory between two moving points in field-free space. The first phase is spent accelerating away from the first body while the terminal phase is spent decelerating into the target body. A discontinuity in the middle for the field-free case is translated into the transition region of the central force-field case.

Although the absolute value of the primer vector magnitude is not important, the relative change of the primer vector magnitude over the trajectory indicates the relative importance of the thrust at a particular point in the trajectory with respect to meeting the final boundary conditions in minimum time. In other words, the relative magnitude of the primer vector is a measure of the influence of the control on the payoff. Figure 4.1 shows a time history of the nominal primer vector magnitude. It is not surprising that the minimum value occurs in the transition region where the thrust-acceleration vector is swinging rapidly around. If a coast period were allowed, it would occur around the point of minimum primer vector magnitude. As will be made clear by other primer vector plots, the qualitative characteristics of the trajectory and control program are very closely related to the time history of the relative primer vector magnitude. Incidentally, in Fig. 4.1 and the remaining primer vector plots the scale is always arbitrarily chosen for the initial point. While this choice is theoretically completely arbitrary, some choices tend to reduce roundoff errors more than others depending upon boundary conditions.

Figure 4.2 shows the time history of the in-plane and out-of-plane steering angles for the nominal trajectory of Fig. 4.0. Although these are the only control variables generated by the output of the algorithm and may not be the most useful for a guidance system, they may be readily converted

into any set of parameters that sufficiently define the control such as angles with respect to vehicle-planet and vehicle-sun lines of sight.

## Trajectory Updated at 53.2 Days

At 1/5, 2/5, 3/5, and 4/5 of the trip time, the trajectory has been updated and various possible guidance errors assumed. These errors have been assumed simultaneously in the six coordinates of the vehicle in phase space. These errors change the initial boundary conditions as follows:

$$r' = (1+\epsilon)r;\ \phi' = \phi + \pi\epsilon_2;\ \theta' = \theta + 2\pi\epsilon_3 \qquad (5.1)$$

$$\dot{r}' = (1+\epsilon_4)\dot{r};\ (r\dot{\phi})' = (1+\epsilon_5)\, r\dot{\phi};\ \ (r \sin\phi\dot{\theta})' = (1+\epsilon_6)\, r \sin\phi\dot{\theta} \qquad (5.2)$$

where the $\epsilon$'s represent the errors, the unprimed terms denote the spherical coordinates and velocity components of the previous nominal and the primed terms denote the corresponding quantities including the error.

For comparison, Fig. 5.0 shows the nominal trajectory updated at 53.2 days. For the trajectory of Fig. 5.1, the following guidance errors were assumed:

$$\epsilon_1 = -0.01\%,\ \epsilon_2 = +0.01\%,\ \epsilon_3 = \epsilon_4 = \epsilon_5 = \epsilon_6 = -0.01\%$$

Comparison of Fig. 5.1 and Fig. 5.0 shows that these guidance errors produce no perceptible change in the control or trip time. In the following figures the errors are increased, always keeping the magnitudes of all components equal and the sign sequence unchanged. The errors indicated in the figures are always with respect to the original nominal.

Figures 5.2 through 5.4 show the effects of 0.1, 1.0, and 2.0% errors, respectively, on the trajectory and control program. The minimum trip time increases from 266.5 days for 0.100% errors to 278.3 days for 2.0% errors. The position errors show up in this sequence of figures in that the initial point of the trajectory is moving back and in with respect to the mark on the Earth's orbit indicating Earth's position at 53.2 days. The steering program remains qualitatively the same although a perceptible change can be seen in the figures, especially in the transition region.

Figure 5.5 shows a plot of the relative primer vector magnitude for the nominal and for the 2.0% error. It is seen that the nature of the trajectory does not change even for errors as large as 2.0% at 53.2 days. The position of minimum primer vector remains essentially the same.

The changes in the in- and out-of-plane control angles are shown in detail by Figs. 5.6 and 5.7 for 0, 1.0, and 2.0% errors. Although not readily detectable in the trajectory figures, the required control changes are easily seen in these two plots. Particularly noticeable is the growing spike required in the out-of-plane control in the transition region as the errors increase.

In summarizing the results for 53.2 days, it can be stated that large errors result in only small increases in minimum trip time and small changes in the control program except possibly for those changes required in the transition region. This basic situation is believed to prevail, more or less, for all updating points before the minimum primer vector point on the nominal trajectory.

Trajectory Updated at 106.4 Days

The results for 106.4 days partially support the statement made above depending upon what is meant by large errors. Again for comparison, Fig. 6.0 shows the original nominal updated at 106.4 days. Figures 6.1 and 6.2 show the trajectories and control programs resulting from 0.01 and 0.10% errors respectively. Neither set of errors result in much of a change in control and the increase in minimum trip time is less than 0.05 days. For 1.0 and 2.0% errors, however, the changes are significant as shown by Figs. 6.3 and 6.4. The minimum trip time increases to 270.9 and 280.7 days respectively. Also, the transition region is moved back in time and the thrust-acceleration vector swings around counterclockwise instead of clockwise.

Figure 6.5 shows the relative primer vector magnitude for 0 and 2.0% errors. It is seen that the minimum primer vector point has been moved back for 2.0% errors. This fact correlates with the observation of the earlier approach of the transition region.

Figures 6.6 and 6.7 give detailed time histories of the in- and out-of-plane steering angles, respectively. In Fig. 6.6 the change in rotation of the thrust-acceleration vector between the nominal case and the 1.0 and 2.0% error cases is plainly shown. Figure 6.7 shows that, starting from the nominal case, a spike in out-of-plane steering angle grows in the transition region up to 1.0% errors and then dies out again at 2.0%. This same effect can be seen in Fig. 6.3 by observing the short length of the ecliptic-plane projection of the third thrust acceleration vector. A physical explanation of this phenomenon is not readily forthcoming, except to say that it is necessary to satisfy the new boundary conditions.

## Trajectory Updated at 159.6 Days

After the point of minimum primer vector magnitude on the nominal trajectory which occurs at about 2.5 $\tau$'s = 145.5 days, the updated trajectory and control become very sensitive to guidance errors. Whereas in the initial phase of the trajectory errors as large as 0.10% did not cause appreciable changes to occur, in the terminal phase errors as small as 0.0005% can cause very large changes in the control as will be demonstrated subsequently.

This fact is not surprising if a corresponding one-dimensional field-free case is analyzed. Suppose that a constant thrust-acceleration vehicle is directly approaching the target body with a relative velocity given by the relation $v = a(T-t)$ and relative position by $r = \frac{1}{2} a(T-t)^2$. In this case, the control program is simply a rearwardly directed constant thrust acceleration. Such a program results in a linear decrease of velocity with time approaching zero velocity at final time T when the relative position is also zero. Eliminating time, the above relations can be expressed by the single equation $v^2 = 2ar$. Now suppose that due to a guidance error the value of $v^2$ is very slightly less than 2ar. It is evident that in this case at the initial point in the trajectory a small amount of forward thrust acceleration will be required to increase the velocity to satisfy the relationship and thus to achieve rendezvous in minimum time. Hence, a very small guidance error would produce a large change in the control at the initial point.

Consider then the other case of $v^2$ very slightly greater than 2ar. In this case the vehicle will go through the target point at a finite velocity even though the thrust acceleration is directed backward the whole way. Nothing can be done to prevent this in this one-dimensional case. If the backward thrusting program is maintained, at some point just slightly beyond the target the relative velocity will be reduced to zero. Now the problem has been reduced to that of rendezvous in minimum time between two points at rest with respect to each other. The proper control program is obviously to maintain the thrust acceleration in the same direction until half the distance has been traversed and then turn it around in the opposite direction in order to come to rest at the target point. In this second case, a very small guidance error causes a large change in control at the final point of the trajectory.

Getting back to the central force-field case at hand, there are a number of factors which cause the situation to be more complicated than the simple case just discussed:

1. The thrust acceleration is not constant but linearly increases with time;

2. The trajectory and control have three degrees of freedom instead of one; and,

3. The small difference in the gravity force on the target and vehicle causes a warping of the stright-line relative trajectory of the field-free case.

These factors, however, are not important enough to destroy a qualitative similarity between the two cases. This similarity will be readily apparent in the results yet to be presented.

Again for comparative purposes, the original nominal trajectory is shown updated at 159.6 days with zero error in Fig. 7.0. Figures 7.1 and 7.2 show the results for 0.002 and 0.003% errors. There are no apparent changes. Figures 7.3, 7.4, and 7.5 show the results for 0.0042, 0.0046, and 0.0050% errors respectively. Note the increasing large change in the control at the final point caused by these small errors.

The original nominal trajectory has a value of $v^2/2ar \approx 1$ throughout the terminal region (where v and r are the relative velocity and radius of the vehicle with respect to Mars) and the relative velocity vector is pointed from the vehicle directly toward Mars. Based upon the discussion of the one-dimensional field-free case, therefore, it is suspected that the assumed guidance errors make $v^2/2ar > 1$. Such is found to be the case for $\epsilon_2 = +0.0042$ and $\epsilon_1 = \epsilon_3 = \epsilon_4 = \epsilon_5 = \epsilon_6 = -0.0042$, for example. By far, the most important error is $\epsilon_6$, the error in in-plane circumferential velocity. The quantity $\epsilon_6$, being negative, contributes to an increase of the relative velocity of the vehicle with respect to Mars. Of course, the radius of the vehicle with respect to the sun decreases ($\epsilon_1$ is negative) and since the vehicle is inside Mars orbit the vehicle-Mars distance also increases as a result of these guidance errors. A detailed inspection of the numbers, however, shows that the velocity error dominates, thereby causing $v^2/2ar$ to become greater than its nominal value. (Actually, the nominal value is a little greater than unity since the thrust-acceleration is not constant, but increasing with time.) The fact that the velocity error dominates in the term $v^2/2ar$ is not surprising since the velocity is squared.

The results of the algorithm are in qualitative agreement with the field-free case. The trajectory does not go through the target and then come back as the one-dimensional field-free case was constrained to do (although this situation is not excluded in the mathematics of the central force-field case). The direction of approach to the Martian "massless point" is completely altered as shown by Figs. 7.3, 7.4, and 7.5.

Figure 7.6 shows the resulting time histories of the primer vector magnitudes corresponding to the nominal 0.002, 0.003, 0.0034, and 0.005% errors. It is seen that the character of the thrusting program does not alter for errors less than 0.003% in that the primer vector magnitude is an increasing monotone. Above 0.003% errors, the primer decreases until at 0.005% a new minimum is observed just before the final point. It is

interesting to observe the similarity of the behavior of the final control for 0.005% errors with that of the nominal at the primer vector minimum in the transition phase.

Detailed plots of the time histories of the in- and out-of-plane control angles for 0 and 0.005% errors are shown in Figs. 7.7 and 7.8.

### Trajectory Updated at 212.9 Days

The only change that is observed as the terminal phase progresses is that the sensitivity of the control to guidance errors becomes even more acute. Where 0.005% errors served to change the control significantly at 159.6 days, at 212.9 days only 0.0005% errors suffice to cause a significant change.

Figures 8.0 through 8.3 give the by now familiar sequence of trajectory plots and thrust-acceleration vectors. Again, the same change of control occurs at the end for 0.0005% errors, and it has been confirmed that the errors produce an increase of $v^2/2ar$ from the nominal value.

Figure 8.4 gives the time histories of the primer vector magnitudes associated with the nominal trajectory and those resulting from 0.0001, 0.0002, 0.0003, and 0.0005% errors. Again, the transition from increasing to decreasing monotones is observed. Figures 8.5 and 8.6 show detailed time histories of the in- and out-of-plane control angles for 0 and 0.0005% errors. These plots are very similar to those of Figs. 7.7 and 7.8.

### Trajectory Updated at 200 Days with Errors in Heliocentric Radius Only

A negative error in heliocentric radius only, keeping the same velocity (i.e., $\epsilon_1 = -(\ )$, $\epsilon_2$ through $\epsilon_6 = 0$), produces a decrease in the parameter $v^2/2ar$. The one-dimensional field-free analogue indicates that there should be a large change in the control in the initial section of the trajectory. Figure 9.0, which shows the initial in-plane steering angle plotted against the percent change in heliocentric radius, verifies this prediction. A heliocentric radius decrease of 0.01% produces a change in angle of about 80 deg. That the radius decrease affects only the initial steering angles is shown by Fig. 9.1 which gives the time histories for 0, -0.006, and -0.010% heliocentric radius errors.

An attempt was made to correlate the results presented by Figs. 9.0 and 9.1 quantitatively with nondimensional plots presented in Ref. 3, which treats the problem of terminal guidance in field-free space for a constant thrust-acceleration vehicle. A satisfactory correlation was not obtained. Reference 4, however, does predict a high sensitivity of the control with respect to the initial boundary conditions for $v^2/2ar < 1$.

## PLANETOCENTRIC NOMINAL TRAJECTORIES

The effort in planetocentric trajectory analysis did not progress to the guidance stage, but was necessarily limited to attempts to establish a nominal minimum-time trajectory with the associated control.

If the low-thrust guidance problem were treated in two distinct parts, involving a planetocentric phase where only the effect of the planet is included, and an interplanetary phase where only the gravity field of the sun is considered, the nominal trajectory could be established as follows: First, a somewhat arbitrary spherical surface is defined (sphere of influence) on which the two trajectories are to be joined. Next, the minimum-time interplanetary trajectory is determined, not with orbiting points to represent the planets as in the preceding section, but with boundary conditions on the spherical surfaces surrounding the planets. The boundary conditions should include transversality conditions which express the desirability that the trajectory pass through the optimum point on the spherical surface with a fixed speed in the optimum direction. Since the two trajectories are treated separately, a parametric search is necessary to determine the optimum speed for a given propulsion system.

The boundary conditions for the planetocentric trajectory are a given initial position and velocity and the final position and velocity as dictated by the interplanetary trajectory. The initial boundary conditions would generally correspond to a point in a low-altitude circular orbit with a velocity increment due to a preliminary high-thrust impulse. If the initial boundary conditions corresponded to circular velocity, with no initial high-thrust velocity increment, the trajectory would cover many revolutions and would require considerable time since the low-thrust thrust-to-weight ratio is of the order of $10^{-4}$. Of course, if the initial high-thrust velocity change is $\sqrt{2}$ times the original circular velocity, the trajectory will be an escape parabola which will not be modified much by the use of low thrust since the time would be short for this case.

### Method of Establishing Nominal Trajectories

For Earth, the planetocentric trajectory should include a radius ratio of about 1/120, i.e., the spherical surface on which the planetocentric and heliocentric trajectories are matched should have a radius of about 120 times that of a low-altitude circular orbit. The sequence for establishing a nominal minimum-time trajectory between these two radii proceeds as follows:

1. An escape parabola is computed with a periradius of 1.0 and a perivelocity of 1.414 where these numbers are in units of the radius and velocity of the initial circular orbit. Position and

velocity vectors are determined at 500 points, equally spaced in time along this parabola from the periradius to the given final radius R, corresponding to 500 mesh points used in the algorithm.

2. The coasting parabola is used as a starting approximation for a variable-thrust trajectory between the two radii with the time specified to be somewhat less than that required by the parabola. In this calculation the transversality conditions,

$$x_i = \frac{R\,\dot{\lambda}_i}{\dot{p}} \qquad i = 1,\ 2,\ 3 \tag{6.1}$$

and

$$\lambda_i = 0 \qquad i = 1,\ 2,\ 3 \tag{6.2}$$

are employed where the $x_i$ are the final position coordinates, the $\lambda_i$ are the corresponding adjoint variables (components of the primer vector), p is the primer vector magnitude, and R is the given final radius. These transversality conditions analytically express the fact that, at this stage, only the final radius is specified while the direction of the final radius vector and the final velocity are completely open and are to be optimized.

3. The variable-thrust trajectory now serves as a starting approximation for a constant-thrust trajectory with the same boundary conditions. It is necessary that the values of initial thrust acceleration and mass flow rate specified for the constant-thrust trajectory be compatible with the time specified for the variable-thrust trajectory.

4. Next, the initial velocity is decreased in steps from the original 1.414 down to the desired value between 1.0 and 1.414. In this sequence, the converged trajectory and adjoint variables for the previous initial velocity serve as a starting approximation for the next. Because the direction of the final position vector is always open, the end point of the trajectory swings around as the initial velocity is reduced and the trajectory winds up like a spring.

5. At this point there exists in the computer a coplanar constant-thrust minimum-time trajectory with the prescribed initial boundary conditions, but it probably does not reach the proper final boundary conditions. A final tracking sequence is required to change the final boundary conditions to the prescribed values.

Two examples from the sequence described in (4) above are shown in Figs. 10.0 and 10.1 for a nominal radius ratio of 1 to 30. (A ratio of 1 to 120, previously indicated to be required for Earth escape, proved to be unattainable due to the difficulty of accommodating enough mesh points to adequately describe the trajectory.) The assumed initial thrust-to-weight ratio for the sample problem is 0.0015. The initial tangential velocity is 1.35 in Fig. 10.0 and has been reduced to 1.34 for Fig. 10.1. This small reduction in initial velocity causes the final point to move counterclockwise approximately 30 deg and the time to increase from 176.3 to 224.9 (in units of the time it takes a point in the initial circular orbit to move through one radian).

## Machine Storage Problems

The procedure outlined above and illustrated by Figs. 10.0 and 10.1 is limited in practice by the storage capacity of the computer. In fact, the 500 mesh points employed in the algorithm to produce the trajectories of Figs. 10.0 and 10.1 practically fill the entire core storage capacity of an IBM 7094 computer.

Another problem is that the mesh points are evenly spaced in time. Since the vehicle is moving very rapidly at the beginning and slowly at the end, this situation results in very low mesh-point density initially, where the curvature is largest, and a high mesh point density toward the end, where the curvature is least. It might seem that mesh point spacing which is weighted toward the beginning of the trajectory would solve this problem. As the trajectory is wound up into one or more revolutions, however, the radius becomes oscillatory with time. In this situation only a variable mesh point spacing scheme such as the one presented in the next section will suffice.

In conclusion, it can be stated that the finite-difference Newton-Raphson algorithm with the tracking sequence just presented can, in principle, determine low-thrust planetocentric spiraling trajectories, but not with restrictions of both in-core storage and constant mesh-point spacing.

## TIME-VARYING GRAVITY FIELD

The traditional way of analyzing the low-thrust interplanetary trajectory problem is to consider the vehicle to move in a sequence of central-force fields generated by the most influential body at each point in the trajectory and employing patching at spheres of influence or asymptotic matching. Although this method is certainly adequate for purposes of mission analysis, its validity is perhaps open to question for the more demanding functions of guidance and control. Of course, for the majority of time on the interplanetary trip, the influence of the planets is very weak compared to that of the sun and the solar central-force field is an extremely good approximation. However, there exist regions in the vicinities of the initial and target planets where the respective gravity accelerations due to the sun and planet are of the same order of magnitude. An examination of the nominal trajectory of Fig. 4.0 will show that at both ends of the trip a considerable amount of time is spent in these regions where the planetary effect is not negligible.

It was pointed out in the last section that the terminal phase of the interplanetary trajectory is a very critical one with small guidance errors corresponding to large required changes in control. Since a large part of the terminal phase takes place in the proximity of the target planet, it is not unreasonable to assume that the inclusion of the planet's gravity field in the problem model would have a significant influence on the resulting control.

For these reasons, the problem was originally formulated for a time-varying gravity field including the contributions from the initial and target planets as well as the sun. All of the bodies were considered to be point masses and the planets were assumed to be in constant Keplerian orbits about the sun.

Two analytical difficulties forced the abandonment of this approach in view of the limited time available for the study. The first was due to orders of magnitude difference between interplanetary and planetocentric distances. The second was the requirement of variable mesh point spacing to insure a high mesh point density in regions having a large gravity field gradient. Before this approach was abandoned, however, the variational equations had been derived, a variable-mesh finite-difference Newton-Raphson algorithm had been coded, and convergence was achieved for a minimum-time constant-thrust trajectory between two sets of boundary conditions which were both well away from any planetary singularities in the gravity field. Despite the fact that the planetary perturbation was weak in the region of the converged trajectory a difference in control was noticed as compared to the corresponding solution for the same boundary conditions without the planetary perturbations. Although this program is still in a development stage, a brief description of the problem formulation will be given.

## Variational Equations

For the case of a time varying gravitational field generated by m point masses whose positions $(x_{1j}, x_{2j}, x_{3j})$, $1 \le j \le m$, are known functions of time, equations (3.13) and(3.12) can be generalized into the following simple form:

$$\ddot{x}_i - \frac{a\lambda_i}{p} + \sum_{j=1}^{m} \frac{\mu_j q_{ij}}{r_j^3} = 0 , \qquad i = 1, 2, 3 \tag{7.1}$$

$$\ddot{\lambda}_i - \sum_{j=1}^{m} \frac{\mu_j}{r_j^3} \left( \frac{3q_{ij} s_j}{r_j^2} - \lambda_i \right) = 0 \quad i = 1, 2, 3 \tag{7.2}$$

where $\mu_j$ is the gravitational constant of the jth body (i.e., point mass), $q_{ij} \equiv x_i - x_{ij}$, $s_j \equiv \sum_{i=1}^{3} \lambda_i q_{ij}$, and $r_j = (\sum_{i=1}^{3} q_{ij}^2)^{1/2}$. $x_i$ is the ith coordinate of the vehicle and $x_{ij}$ is the ith coordinate of the jth gravitating body.

The terminal transversality conditions can take the form of optimal entry into an orbit, or even optimal entry into a parametric family of orbits, about the target body. However, for simplicity in numerical experiments to date, this condition has been fixed as entry into a specified orbit at a specified point. The arbitrary scaling of the primer vector remains valid as in the fixed-field problem.

## Equations for the Numerical Solution

Since the knowledge of the $q_{ij}$ for any j completely defines the system, it is advantageous to choose a particular j, call it $\hat{j}$, and write equation (7.1) with respect to the $q_{i\hat{j}}$. Therefore, define $\ddot{q}_{i\hat{j}} = f_i$ and $\ddot{\lambda}_i = g_i$ where

$$f_i = \frac{a\lambda_i}{p} - \sum_{j=1}^{m} \frac{\mu_j q_{ij}}{r_j^3} - \ddot{x}_{i\hat{j}} , \tag{7.3}$$

and $g_i$ is determined by equation (7.2).

The elements of the Jacobian follow the pattern of equation (3.19)

$$\frac{\partial f_i}{\partial q_{i\hat{j}}} = \sum_{j} \frac{\mu_j}{r_j^3} \left( \frac{3q_{ij}^2}{r_j^2} - 1 \right) ,$$

$$\frac{\partial f_i}{\partial q_{\nu \hat{j}}} = 3 \sum_j \frac{\mu_j q_{i\,j} q_{\nu j}}{r_j^5}, \qquad i \neq \nu,$$

$$\frac{\partial f_i}{\partial \lambda_i} = \frac{a}{p^3} (p^2 - \lambda_i^2), \tag{7.4}$$

$$\frac{\partial f_i}{\partial \lambda_\nu} = - \frac{a}{p^3} \lambda_i \lambda_\nu, \qquad i \neq \nu$$

$$\frac{\partial g_i}{\partial q_{i\hat{j}}} = 3 \sum_j \frac{\mu_j}{r_j^5} \left[ 2 q_{i\,j} \lambda_i - s_j \left( \frac{5 q_{i\,j}^2}{r_j^2} - 1 \right) \right],$$

$$\frac{\partial g_i}{\partial q_{\nu \hat{j}}} = 3 \sum_j \frac{\mu_j}{r_j^5} \left( \lambda_i q_{\nu j} + \lambda_\nu q_{i\,j} - \frac{5 q_{i\,j} q_{\nu j} s_j}{r_j^2} \right), \qquad i \neq \nu$$

$$\frac{\partial g_i}{\partial \lambda_i} = \frac{\partial f_i}{\partial q_{i\hat{j}}}, \quad \text{and} \quad \frac{\partial g_i}{\partial \lambda_\nu} = \frac{\partial f_i}{\partial q_{\nu \hat{j}}}, \qquad i \neq \nu$$

There is no change in the scaling equation (3.18), but equation (3.21) becomes

$$\frac{\partial f_i}{\partial t} = - \frac{a\,b}{(1 + bt)} \frac{\lambda_i}{p} - \dddot{x}_{i\hat{j}} \tag{7.5}$$

## Numerical Problems

It is fairly clear that the nature of the solution is going to change in the vicinity of the different bodies. For this reason, both the ability to change $\hat{j}$ at several points in the solution and the means to handle variable mesh spacing is required. These problems have been attacked, but have not been overcome. Recently, Breakwell and Rauch (Ref. 5) have published a mixed analytical and numerical solution to the three mass-point, variable specific-impulse, optimum-rendezvous problem. This solution is interesting in two respects: First, knowing the general nature of the solution alleviates the numerical problems. Second, the variable specific-impulse problem offers somewhat simpler and more stable equations for numerical experimentation.

The basic approach taken to date has been to choose $\hat{j}$ such that $r_j^2/\mu_j$ is a minimum over all j at every point. In addition, the mesh point spacing is made proportional to $r_{\hat{j}}^2/\mu_{\hat{j}}$. It appears that the basic difficulty of this approach is the severe discontinuity in the gradient of this function at the points where $\hat{j}$ changes value.

## RECOMMENDATIONS FOR FURTHER RESEARCH

The results of the present study have indicated a number of problems which in the authors' opinion warrant further investigation.

It is generally felt that while the present effort has developed an excellent numerical algorithm for use in low-thrust guidance problems, it still remains to exercise the algorithm in a systematic way in order to gain a basic understanding of practical problems of low-thrust guidance and control in a central-force field, especially in the terminal phase. A very general situation such as rendezvous with a point in a circular orbit should be studied and the optimal control from all points in the neighboring region of phase space determined. Perhaps from such a study, simple semi-empirical and/or semi-analytical terminal low-thrust guidance laws could be formulated and compared with the results of the numerical algorithm.

One question that is raised by the extreme sensitivity of the control to small guidance errors in the terminal phase is whether low-thrust systems with a very narrow range of available control can achieve rendezvous unassisted. This question is by no means academic.

Since there is no space nuclear-electric powerplant currently being developed of sufficient capacity to provide prime power for an electric propulsion system, the only hope for low-thrust systems in the near future is the use of solar power. Whether solar energy is employed to heat a working fluid or the solar radiation is converted directly into electrical energy, these devices will require large expanses of material to intercept the sunlight. Due to the large moments of inertia and stability problems of these vehicles, the thrust-attitude control probability will be severely limited. In fact the current thinking is in terms of constant thrust-attitude control.

It is recommended, therefore, that an interplanetary guidance study be undertaken for a vehicle constrained to direct its thrust throughout a very small solid angle. Such a problem would involve inequality constraints which would be difficult to handle but probably not impossible with a modification of the present algorithm.

Perhaps as a requirement for actual low-thrust guidance, or at least as a complete solution to the realistic problem to which other less sophisticated guidance schemes should be compared, the time-varying field analysis started in this study should be continued.

Finally it is believed that all that is required to solve the planetocentric problem is the use of variable mesh point spacing and storage outside of core. Since machine time would not be of prime consideration in a real-time, on-board computer application, external storage might be feasible.

## REFERENCES

1. Van Dine, C. P., W. R. Fimple, and T. N. Edelbaum: Application of a Finite-Difference Newton-Raphson Algorithm to Problems of Low-Thrust Trajectory Optimization. Progress in Astronautics, Vol. 17, Academic Press Inc., New York, 1966.

2. Lawden, D. F.: Optimal Trajectories for Space Navigation. Butterworth's, London, 1963.

3. Kantorovich, L. V. and G. P. Akilov: Functional Analysis in Normed Spaces. Chapter 18, Pergamon Press, New York, 1964.

4. Bryson, A. E.: Nonlinear Feedback Solution for Minimum Rendezvous with Constant Thrust Acceleration. To be published in the Proceedings of the 16th International Astronautical Congress, September 13-18, 1965, Athens, Greece. Polish Scientific Publishers, Warsaw, Poland.

5. Breakwell, J. V. and H. E. Rauch: Asymptotic Matching in Power-Limited Interplanetary Transfers. American Astronautical Society Preprint 66-114.

FIG. 1

# EARTH - MARS TRAJECTORY

## NUMERICAL ACCURACY DETERMINATION

## CONSTANT THRUST MINIMUM TIME

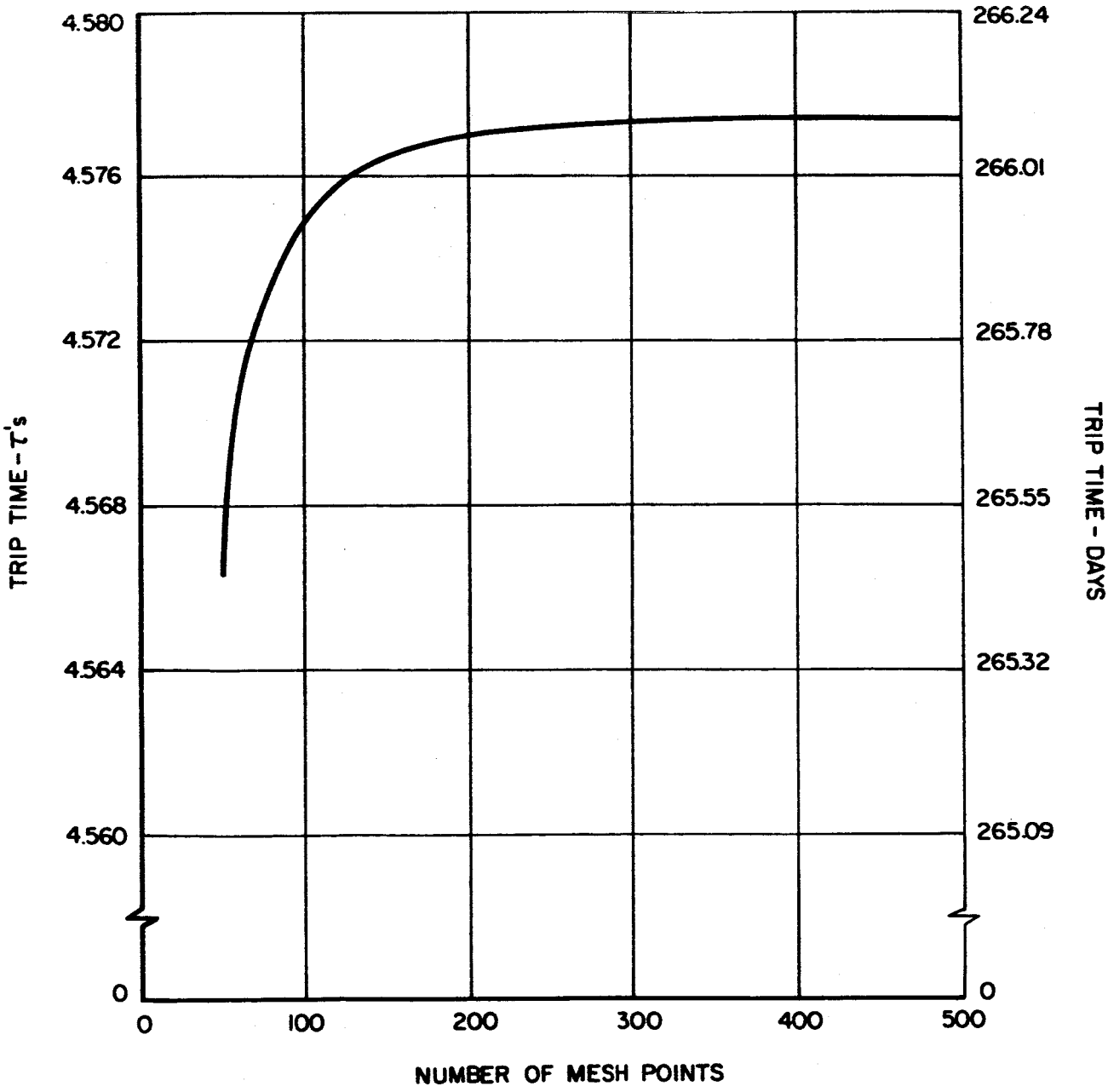


FIG. 2

# EARTH - MARS TRAJECTORY

## NUMERICAL ACCURACY DETERMINATION

## CONSTANT THRUST MINIMUM TIME

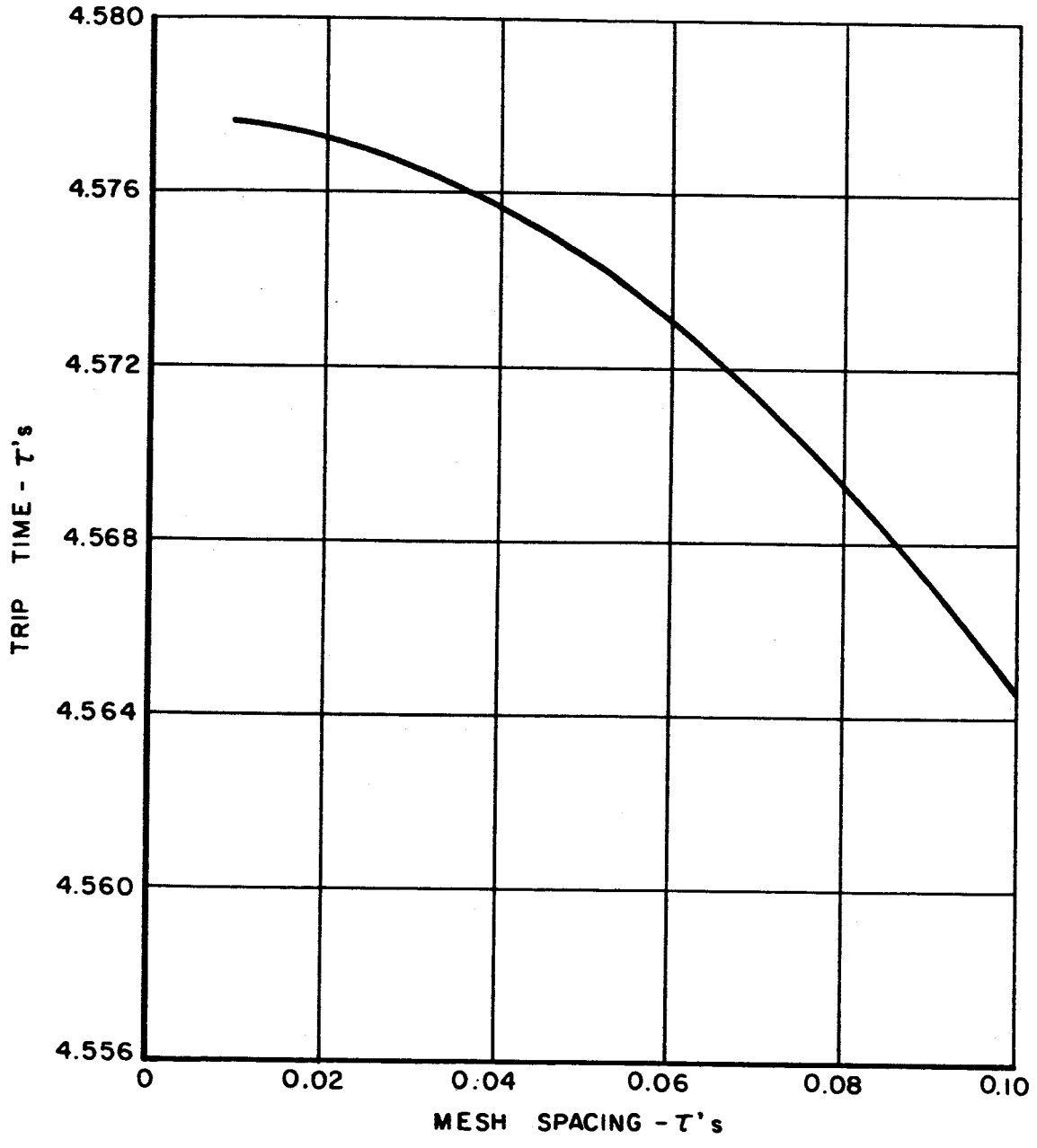


FIG. 3

# EARTH - MARS TRAJECTORY

## NUMERICAL ACCURACY DETERMINATION

## INITIAL STEERING ANGLES

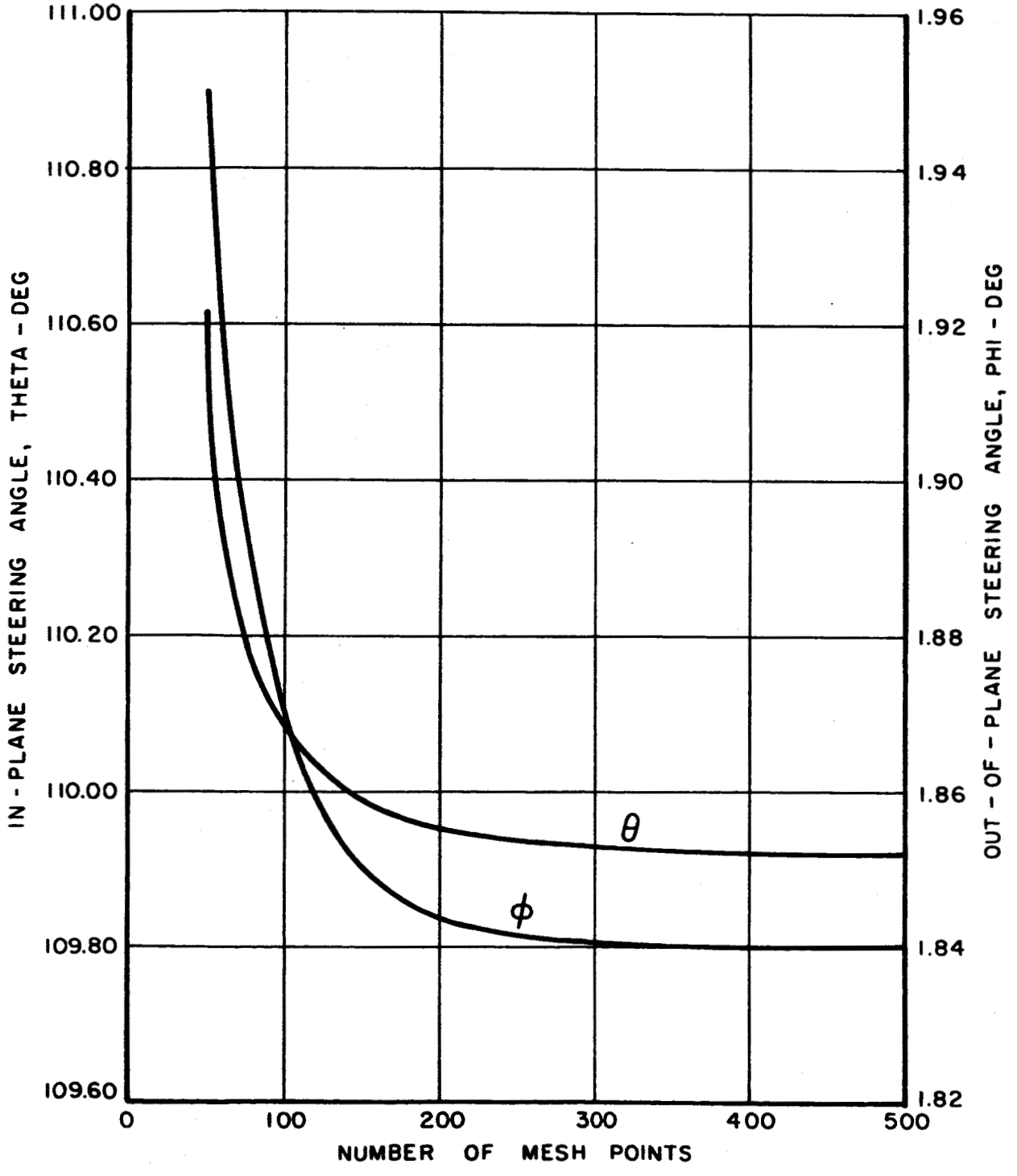


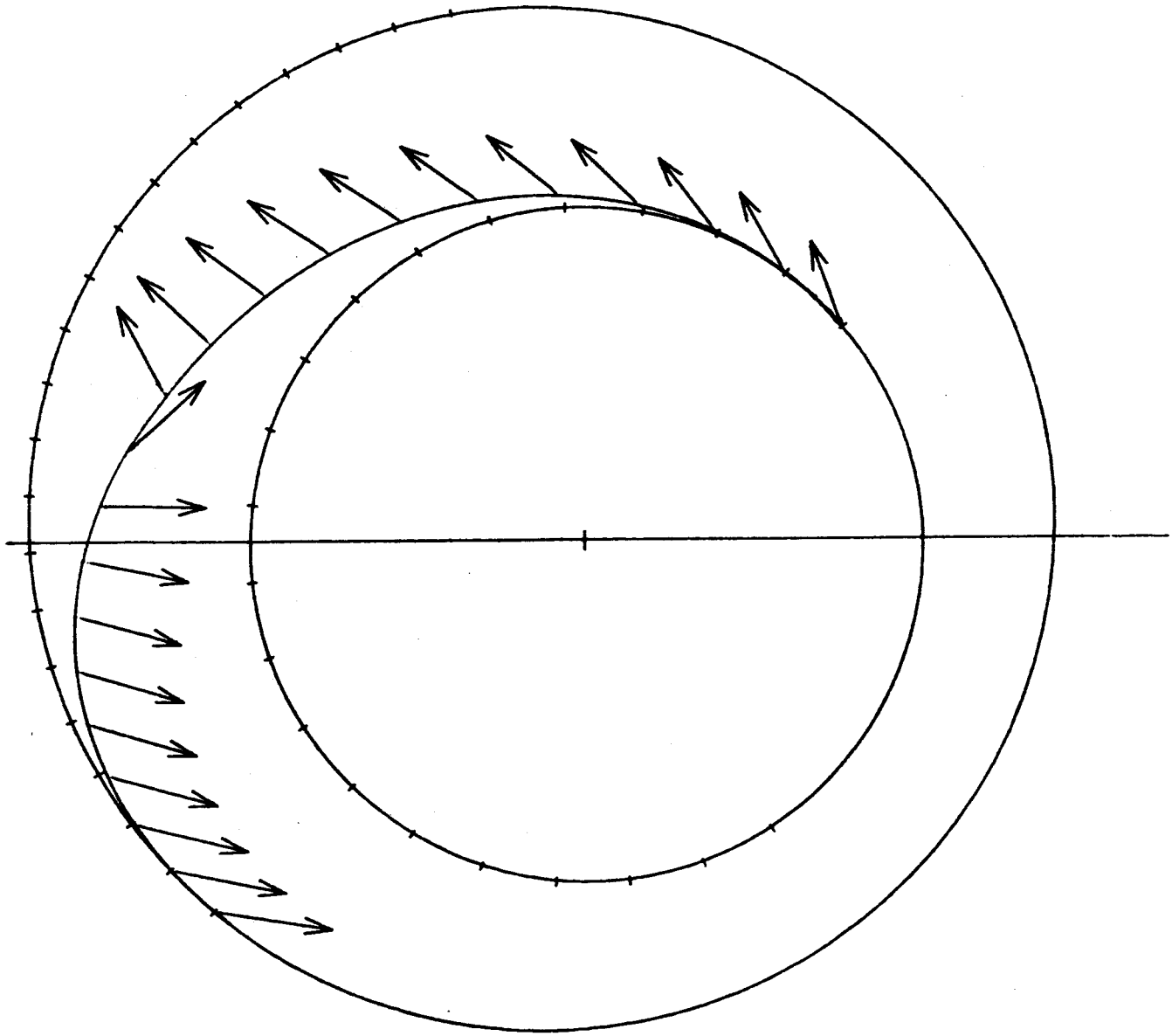
FIG. 4.0

# EARTH - MARS TRAJECTORY

## ECLIPTIC PLANE PROJECTION

## TRIP TIME - 266.1 DAYS

## LAUNCH DATE - 2444180



TIME INTERVAL 13.3 DAYS

⟶ THRUST ACCELERATION

SCALE 0.5 INCH - .001 M/SEC2

FIG. 4.1

# EARTH - MARS TRAJECTORY

## NOMINAL PRIMER VECTOR HISTORY FROM LAUNCH

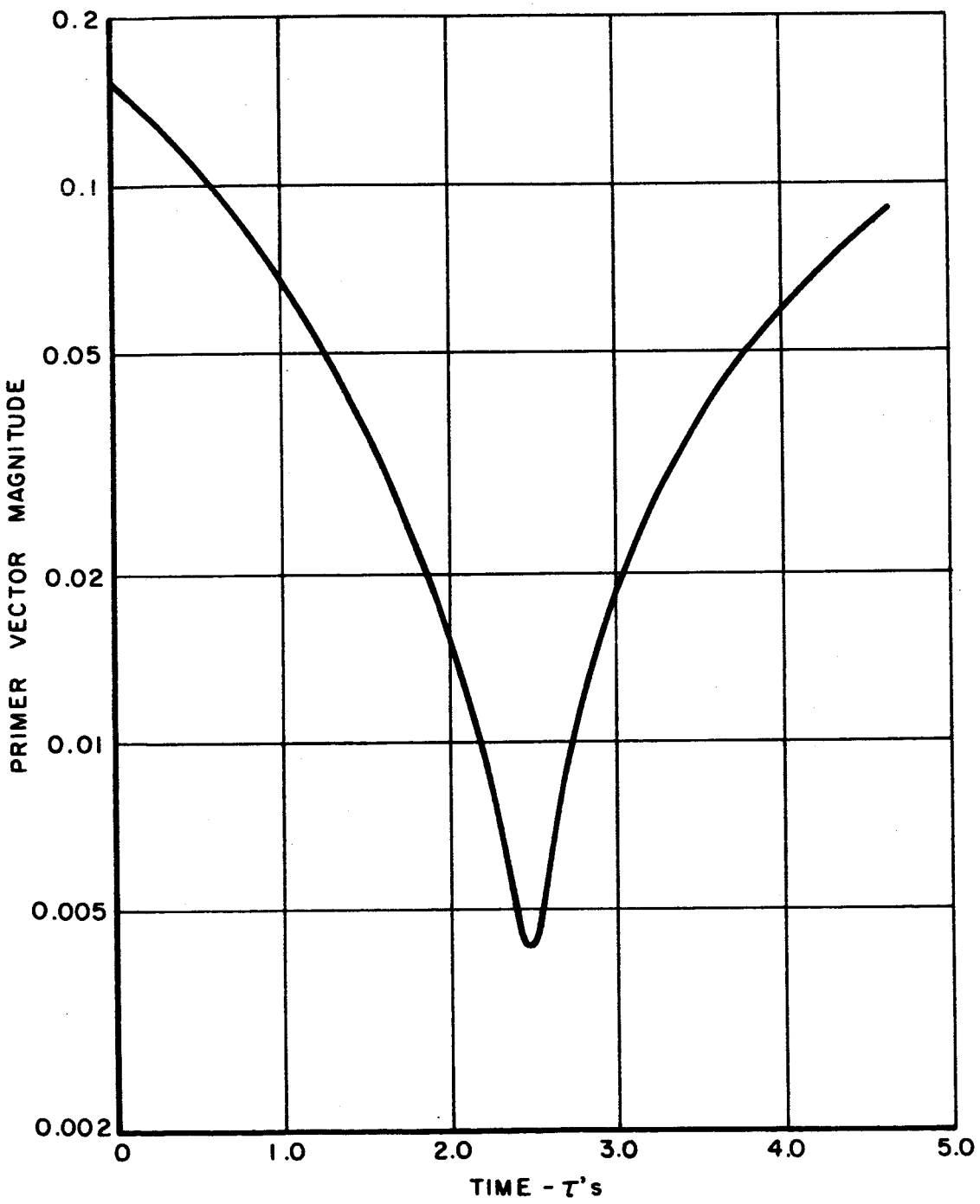


FIG. 4.2

# EARTH - MARS TRAJECTORY

## NOMINAL CONTROL FROM LAUNCH

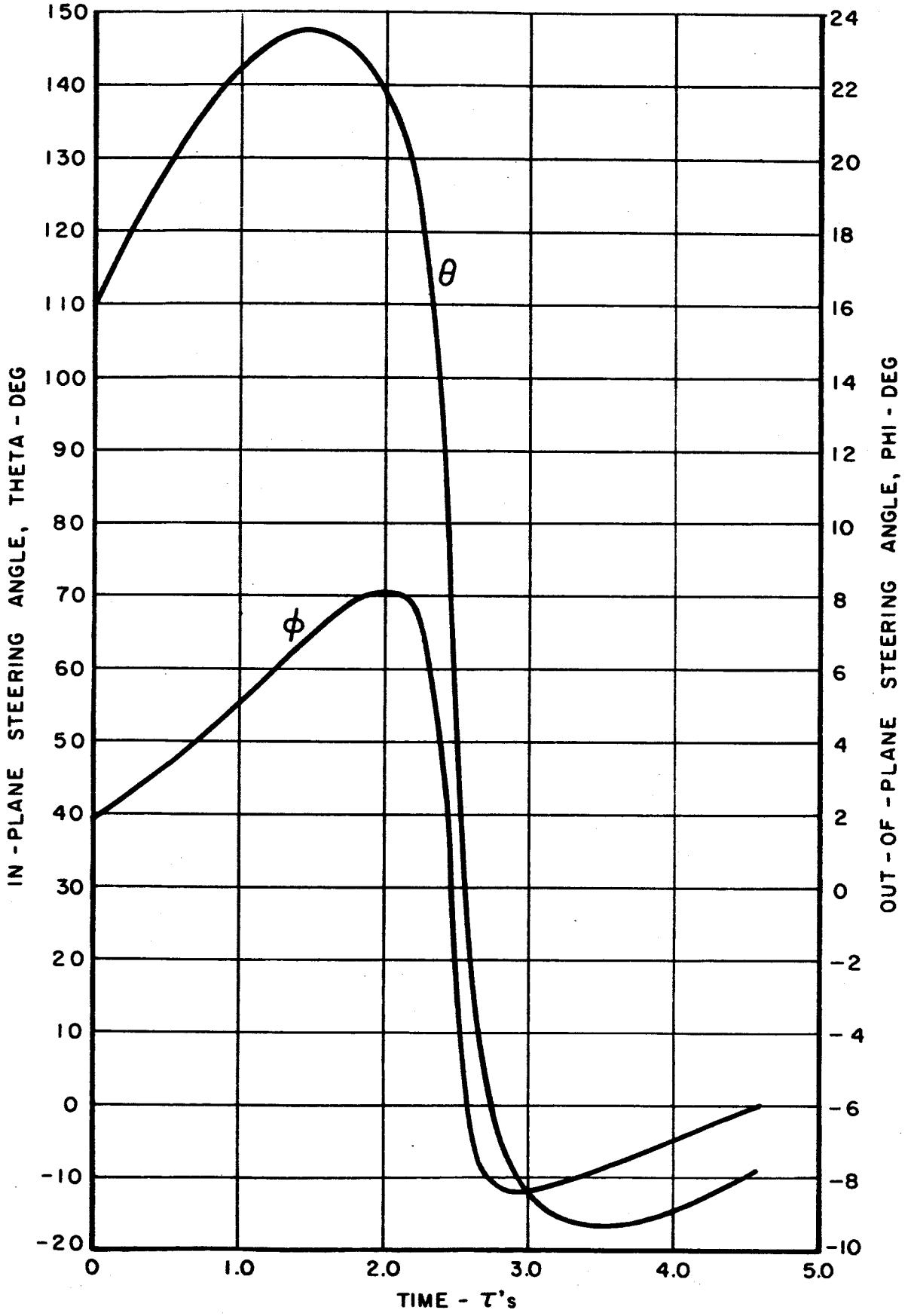


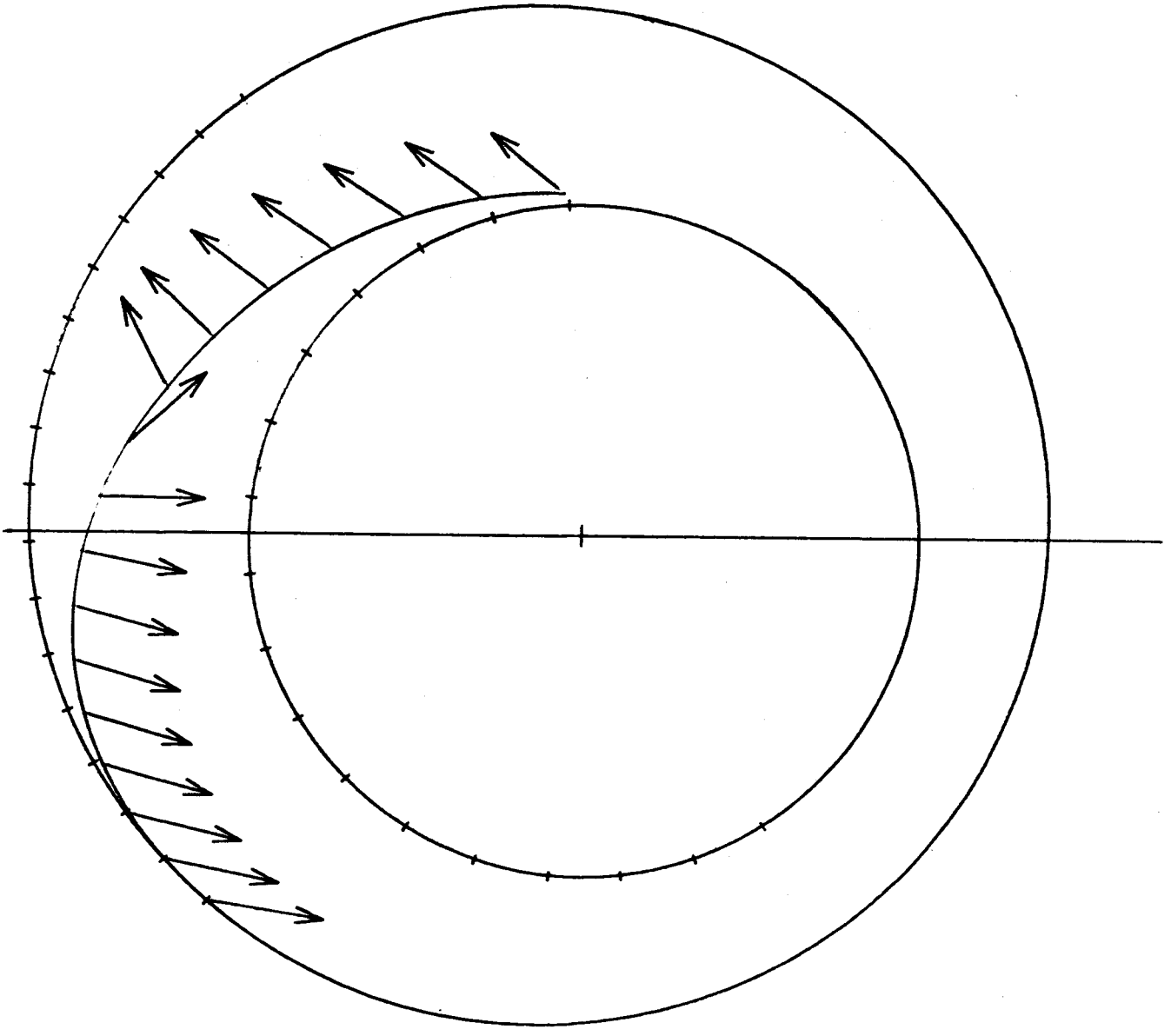
FIG. 5.0

# EARTH - MARS TRAJECTORY

## ECLIPTIC PLANE PROJECTION

## TRIP TIME = 266.1 DAYS

## 0.000 PCT ERRORS AT 53.2 DAYS



TIME INTERVAL = 13.3 DAYS

⟶ THRUST ACCELERATION

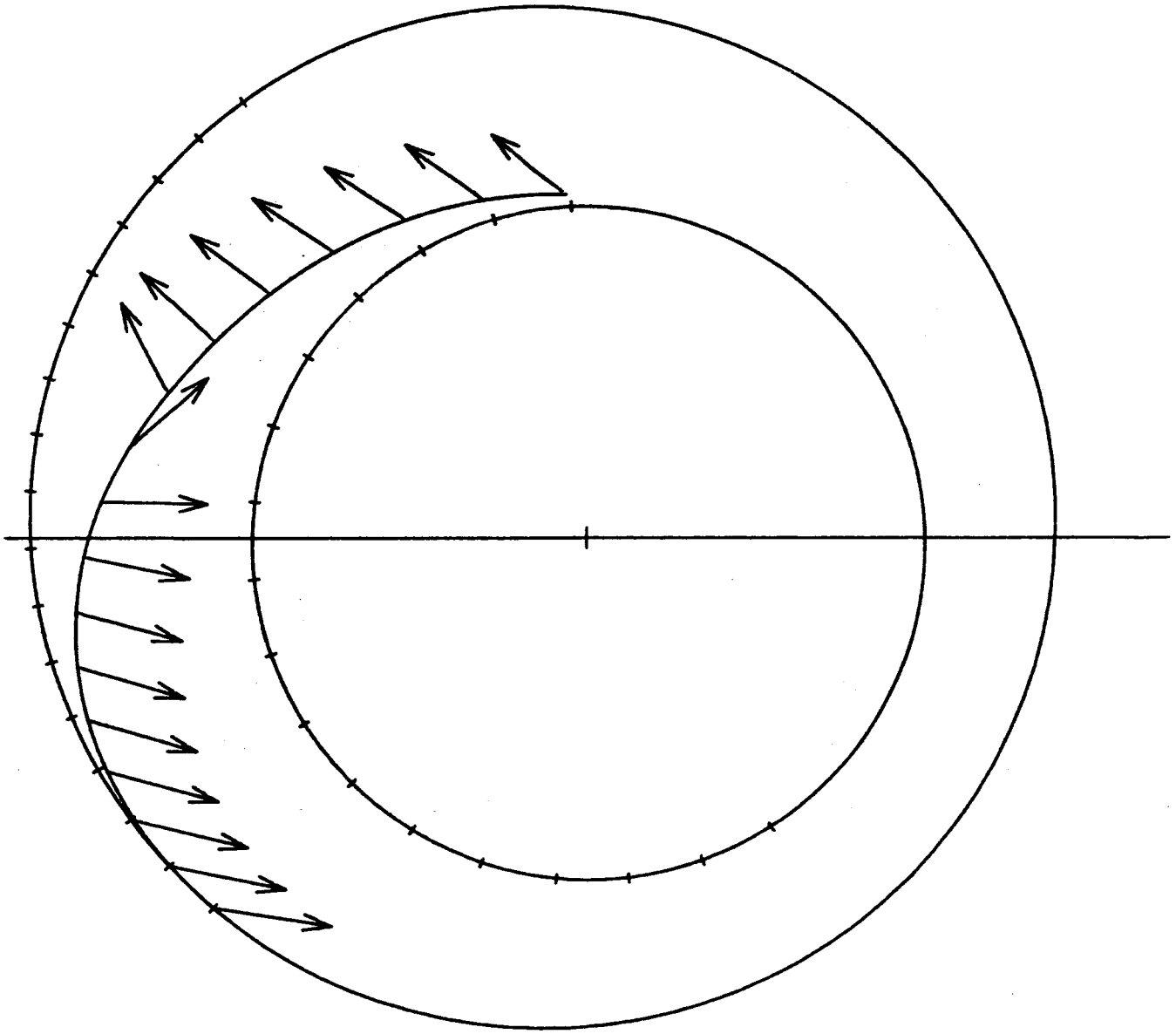
SCALE 0.5 INCH = .001 M/SEC2

# EARTH - MARS TRAJECTORY

## ECLIPTIC PLANE PROJECTION

## TRIP TIME = 266.1 DAYS

## 0.010 PCT ERRORS AT 53.2 DAYS



TIME INTERVAL = 13.3 DAYS

⟶ THRUST ACCELERATION

SCALE 0.5 INCH = .001 M/SEC2

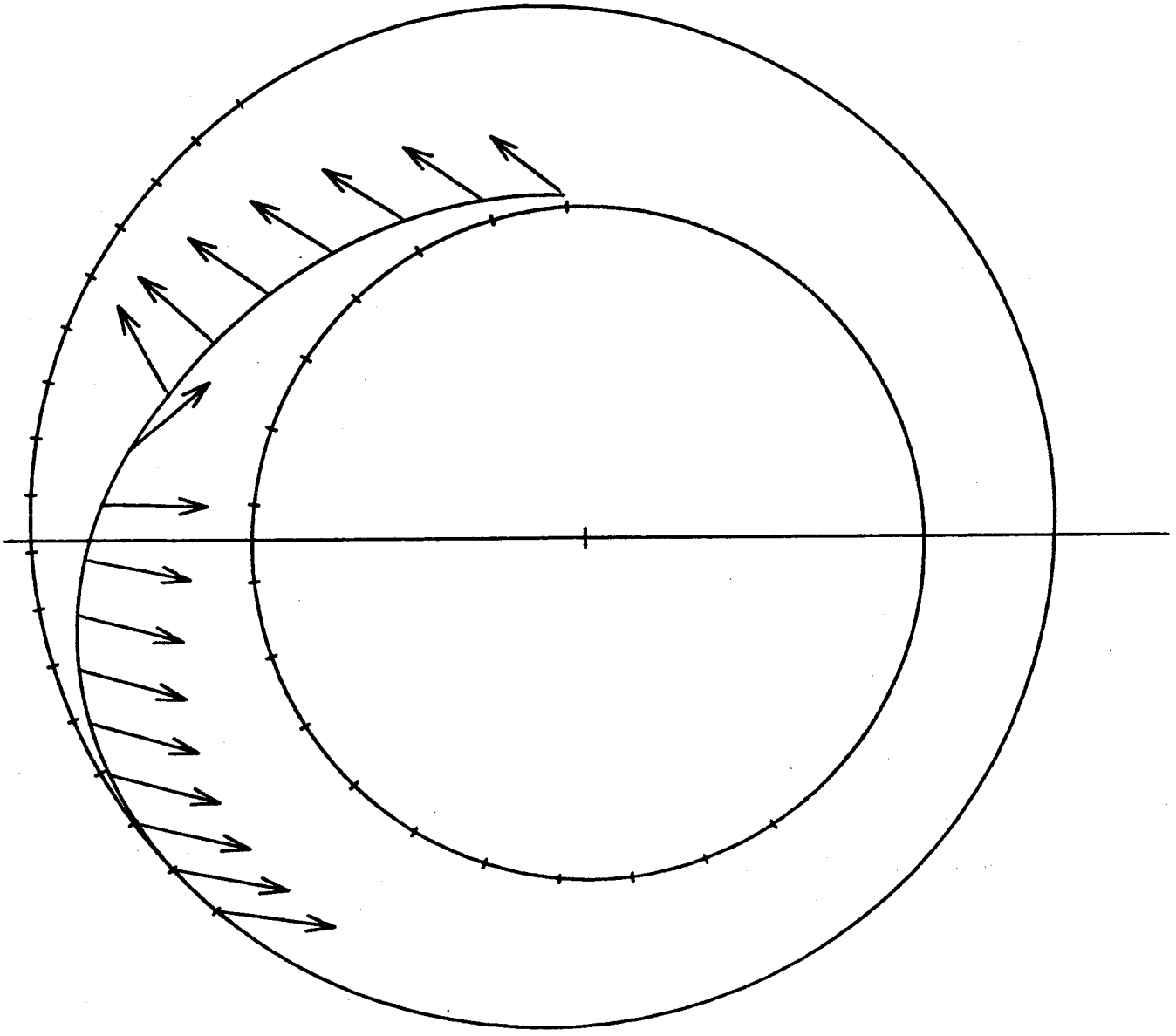
FIG. 5.2

# EARTH - MARS TRAJECTORY

## ECLIPTIC PLANE PROJECTION

## TRIP TIME = 266.5 DAYS

## 0.100 PCT ERRORS AT 53.2 DAYS



TIME INTERVAL = 13.3 DAYS

⟶ THRUST ACCELERATION

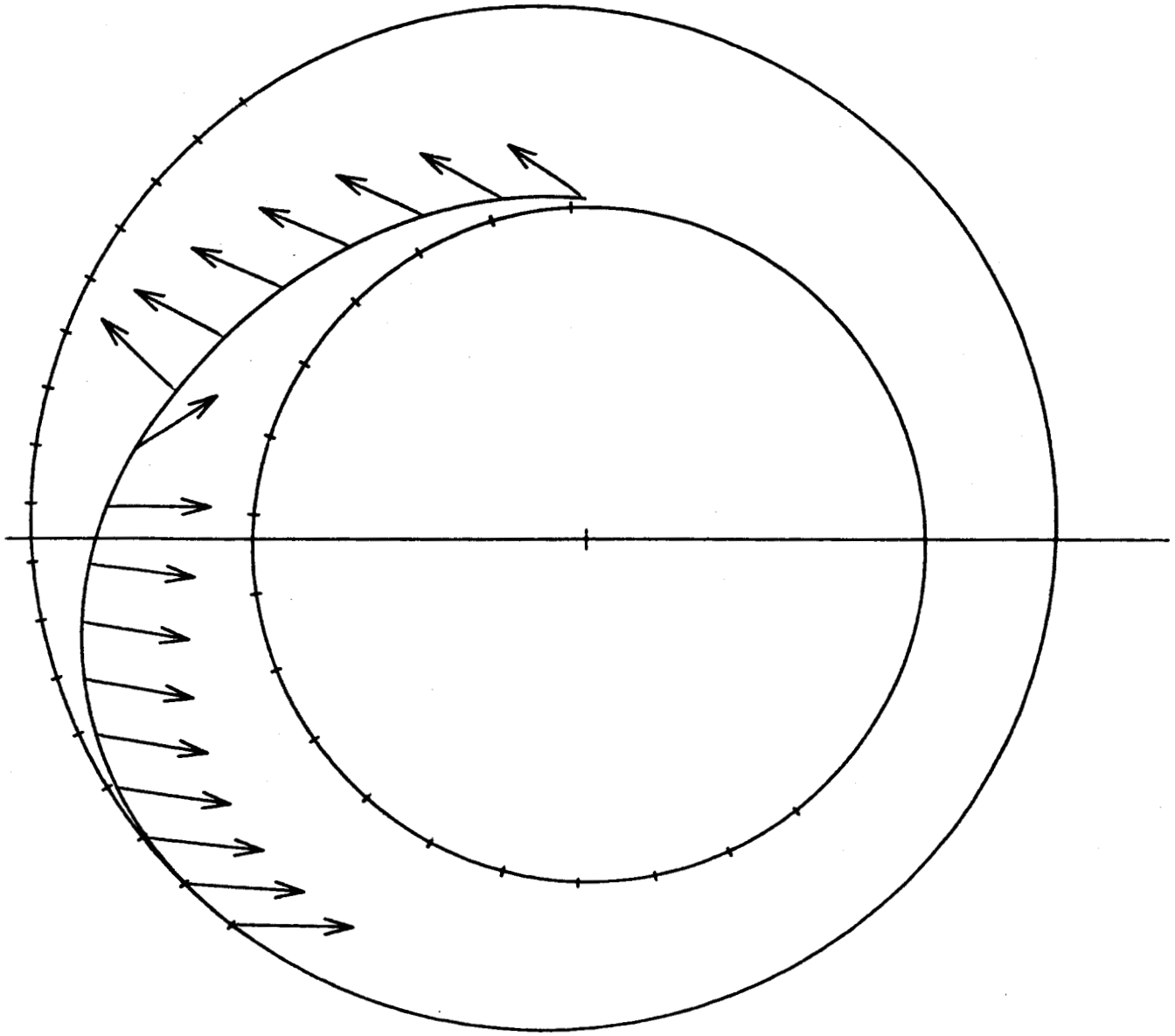
SCALE 0.5 INCH = .001 M/SEC2

# EARTH - MARS TRAJECTORY

## ECLIPTIC PLANE PROJECTION

## TRIP TIME = 271.5 DAYS

## 1.000 PCT ERRORS AT 53.2 DAYS



TIME INTERVAL = 13.7 DAYS

⟶ THRUST ACCELERATION

SCALE 0.5 INCH = .001 M/SEC2

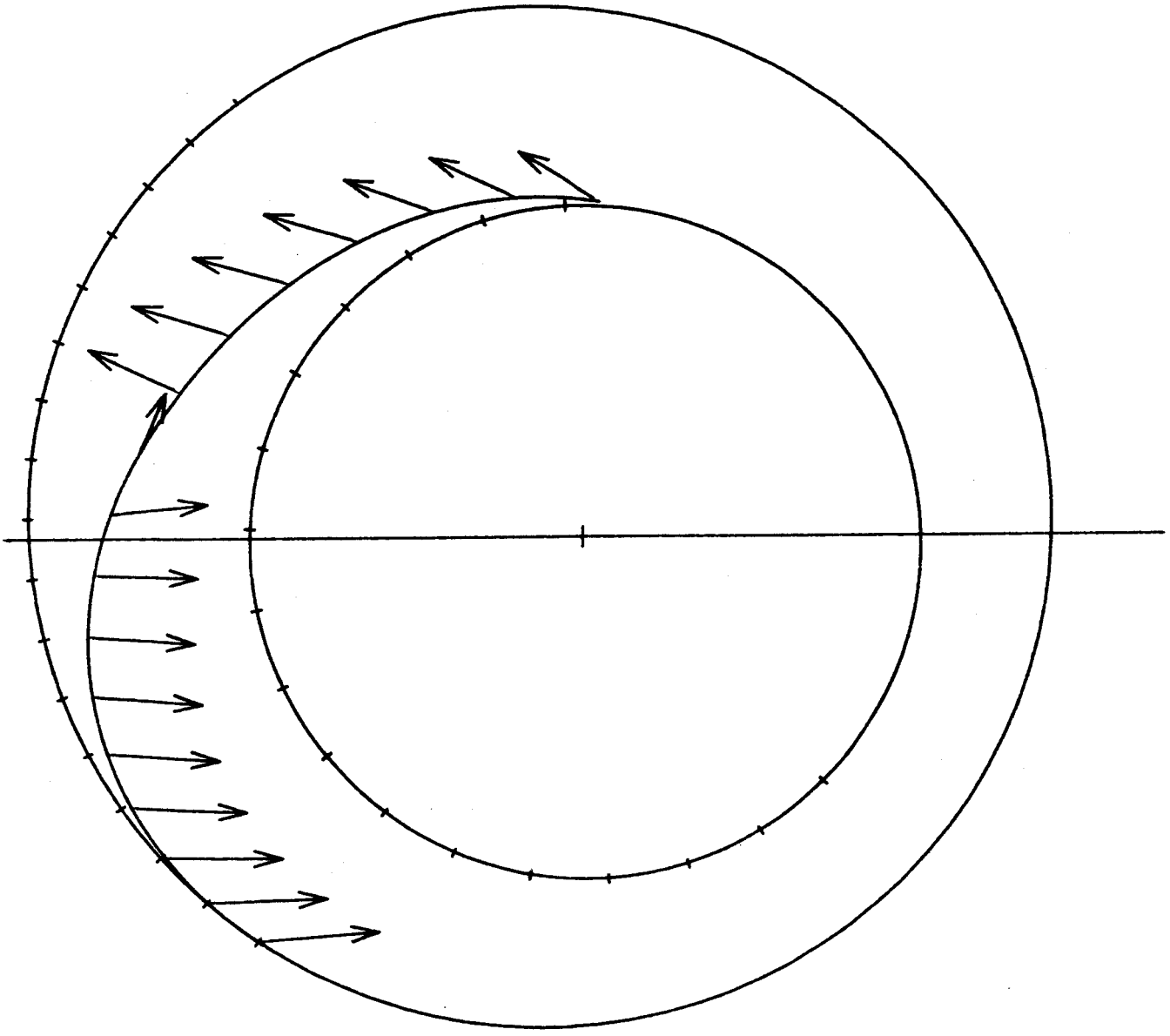
FIG. 5.4

# EARTH - MARS TRAJECTORY

## ECLIPTIC PLANE PROJECTION

## TRIP TIME - 278.3 DAYS

## 2.000 PCT ERRORS AT 53.2 DAYS



TIME INTERVAL - 14.1 DAYS

⟶ THRUST ACCELERATION

SCALE 0.5 INCH - .001 M/SEC2

FIG. 5.5

# EARTH - MARS TRAJECTORY

## PRIMER VECTOR HISTORY FROM 53.2 DAYS

## 0 AND 2 PERCENT GUIDANCE ERRORS

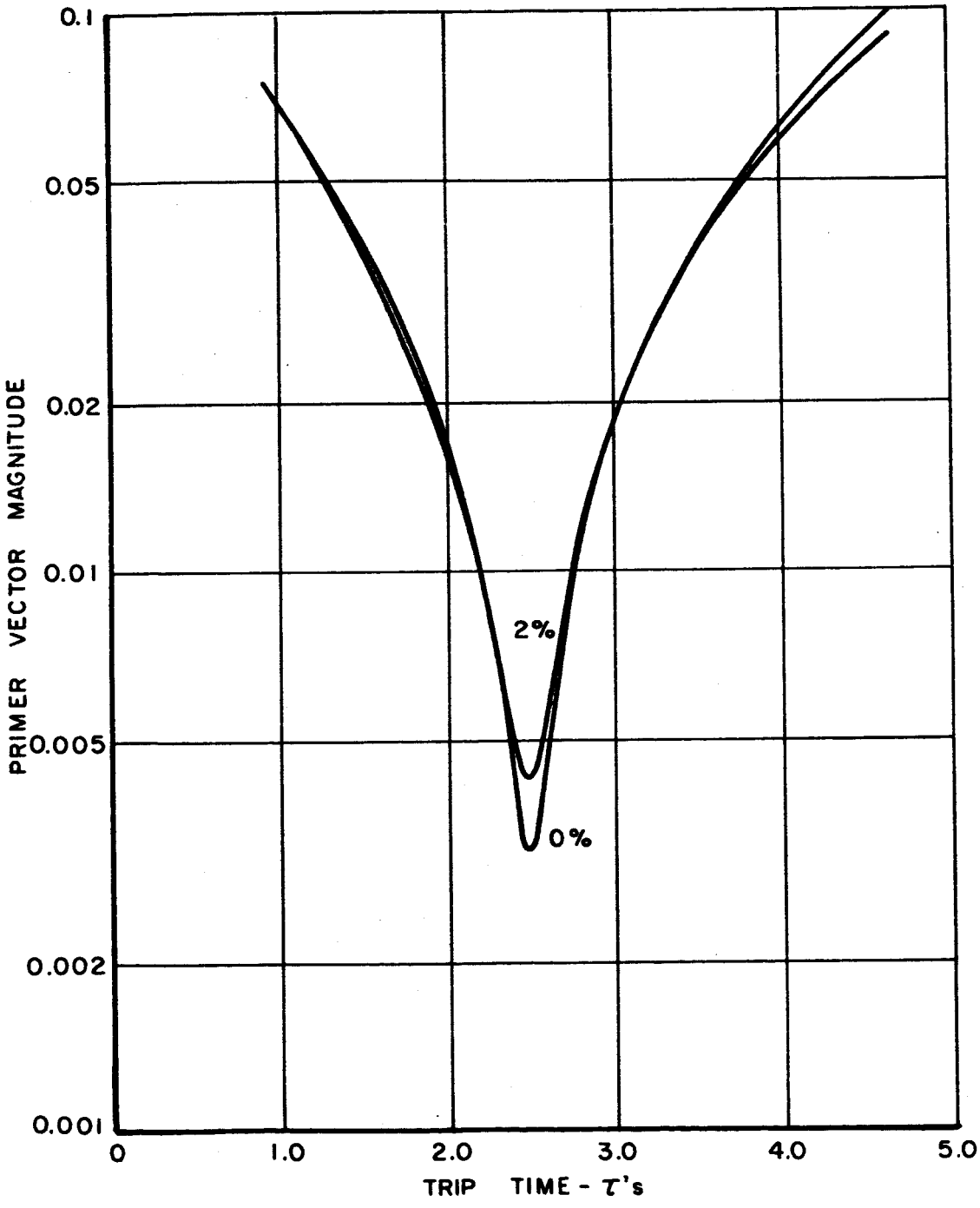


FIG. 5.6

# EARTH - MARS TRAJECTORY

## IN - PLANE CONTROL FROM 53.2 DAYS

## 0, 1, AND 2 PERCENT GUIDANCE ERRORS

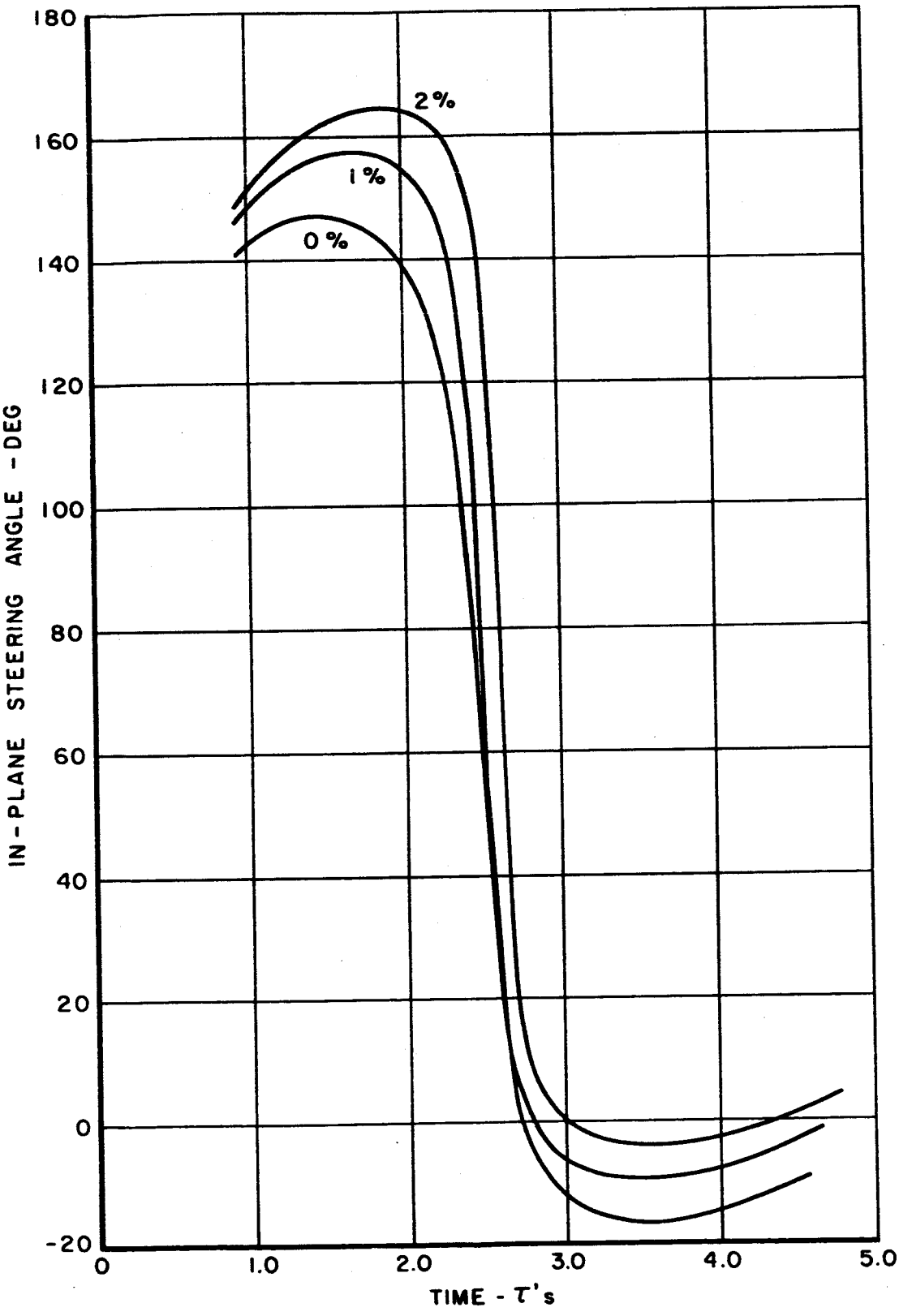


FIG. 5.7

# EARTH - MARS TRAJECTORY

## OUT - OF - PLANE CONTROL FROM 53.2 DAYS

## 0, 1, AND 2 PERCENT GUIDANCE ERRORS

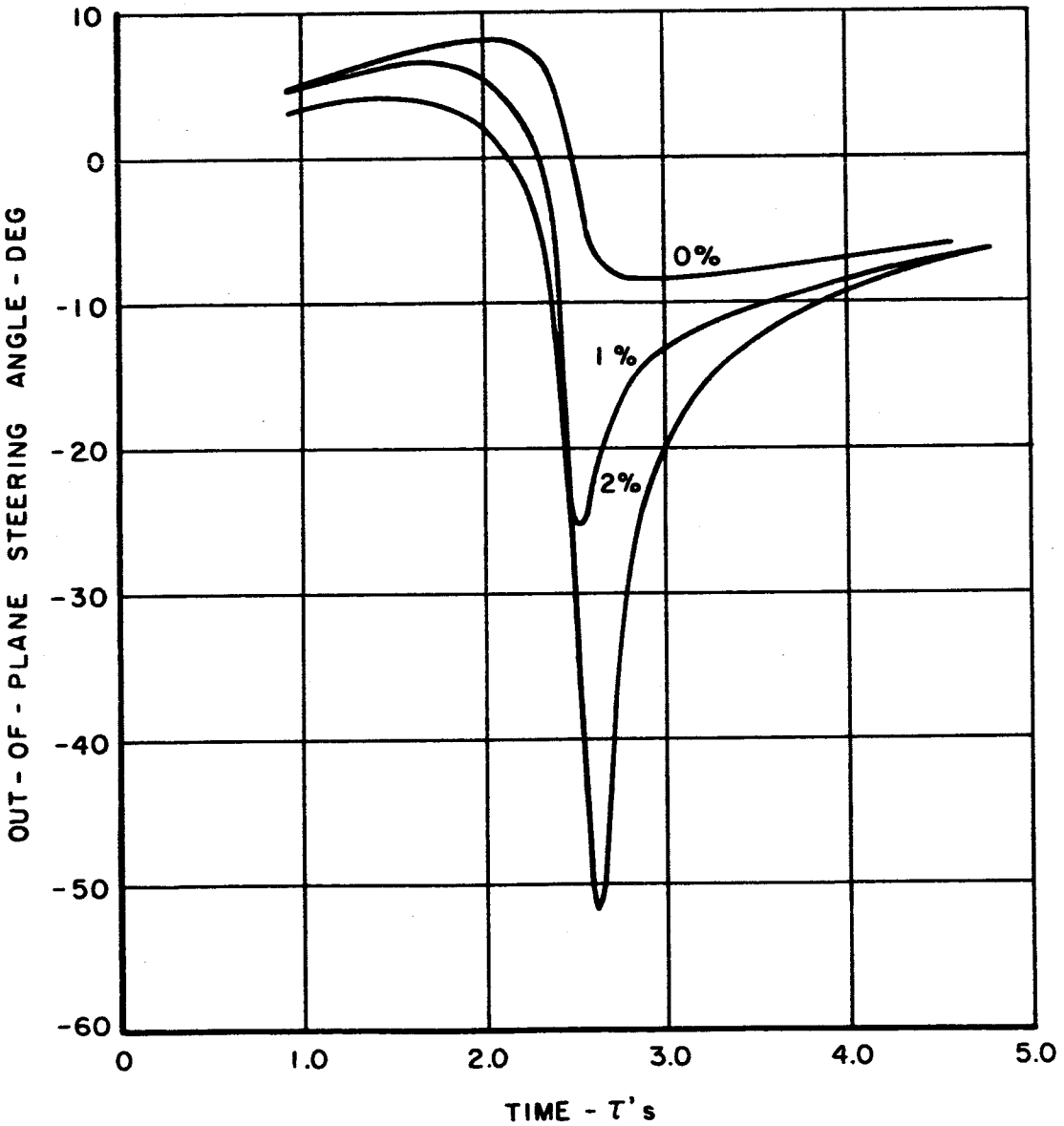


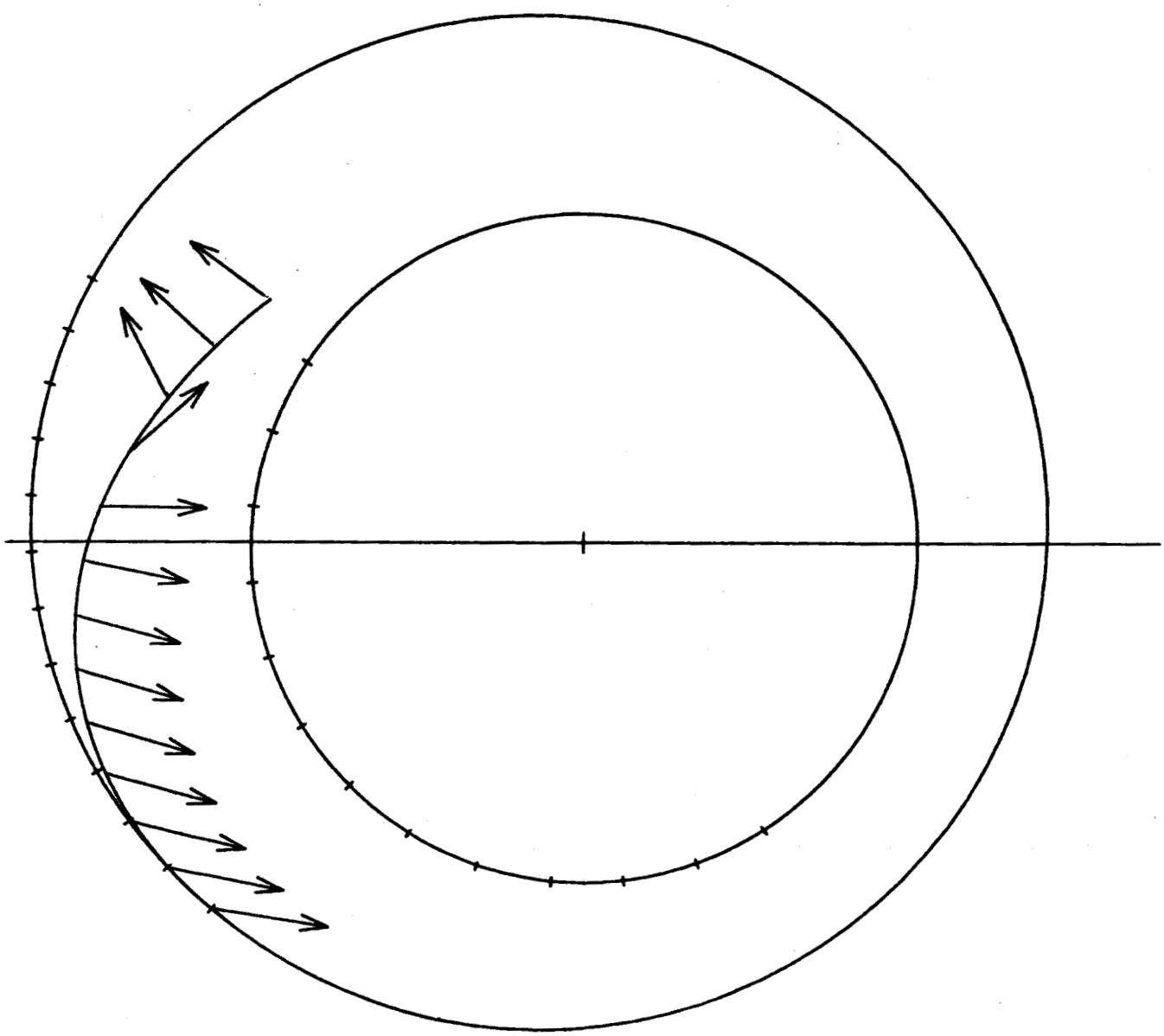
FIG. 6.0

# EARTH - MARS TRAJECTORY

## ECLIPTIC PLANE PROJECTION

## TRIP TIME = 266.1 DAYS

## 0.000 PCT ERRORS AT 106.4 DAYS



TIME INTERVAL = 13.3 DAYS

⟶ THRUST ACCELERATION

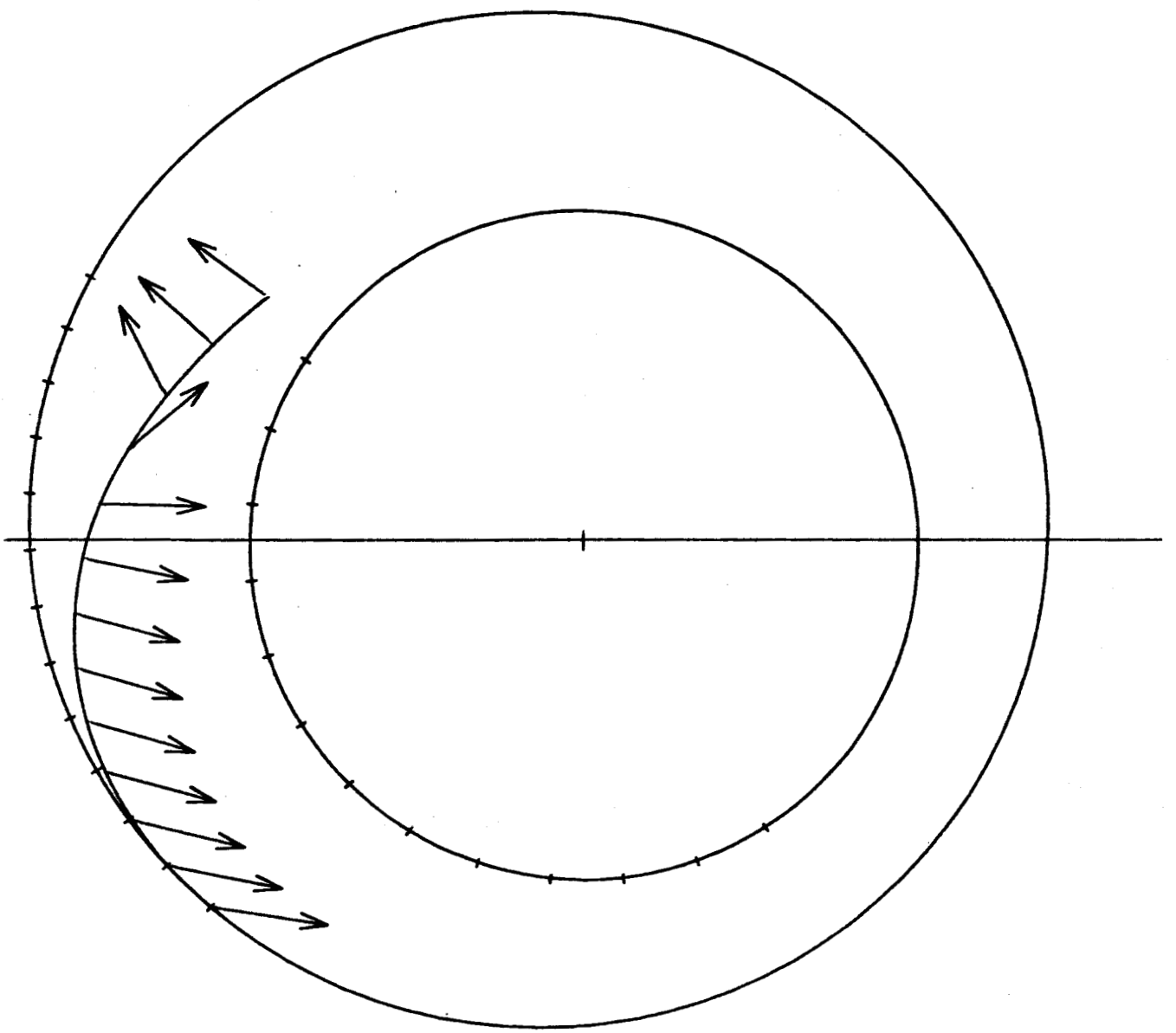
SCALE 0.5 INCH = .001 M/SEC2

# EARTH - MARS TRAJECTORY

## ECLIPTIC PLANE PROJECTION

## TRIP TIME = 266.1 DAYS

## 0.010 PCT ERRORS AT 106.4 DAYS



TIME INTERVAL = 13.3 DAYS

⟶ THRUST ACCELERATION

SCALE 0.5 INCH = .001 M/SEC2

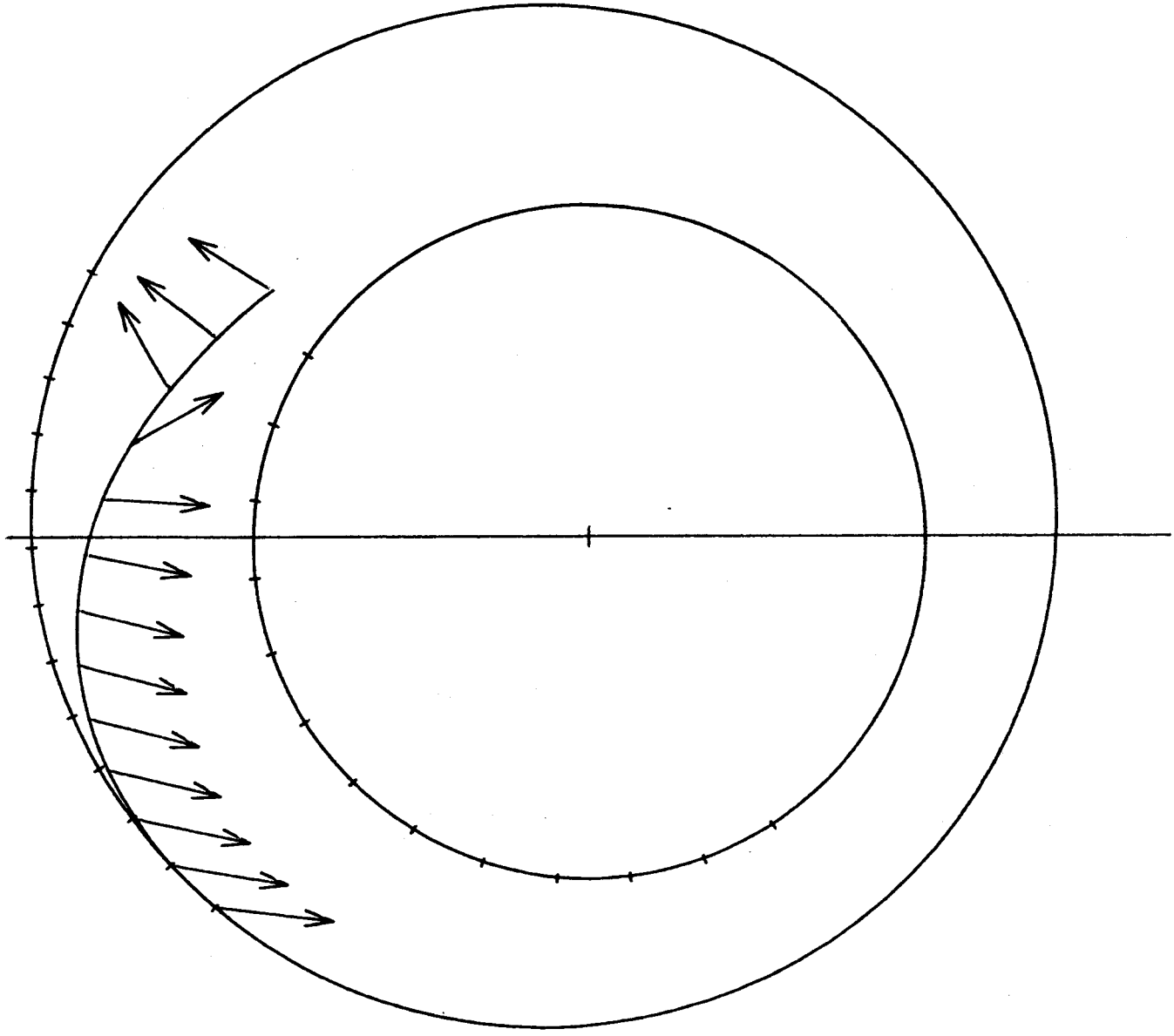
FIG. 6.2

# EARTH - MARS TRAJECTORY

## ECLIPTIC PLANE PROJECTION

## TRIP TIME - 266.1 DAYS

## 0.100 PCT ERRORS AT 106.4 DAYS



TIME INTERVAL - 13.3 DAYS

⟶ THRUST ACCELERATION

SCALE 0.5 INCH - .001 M/SEC2

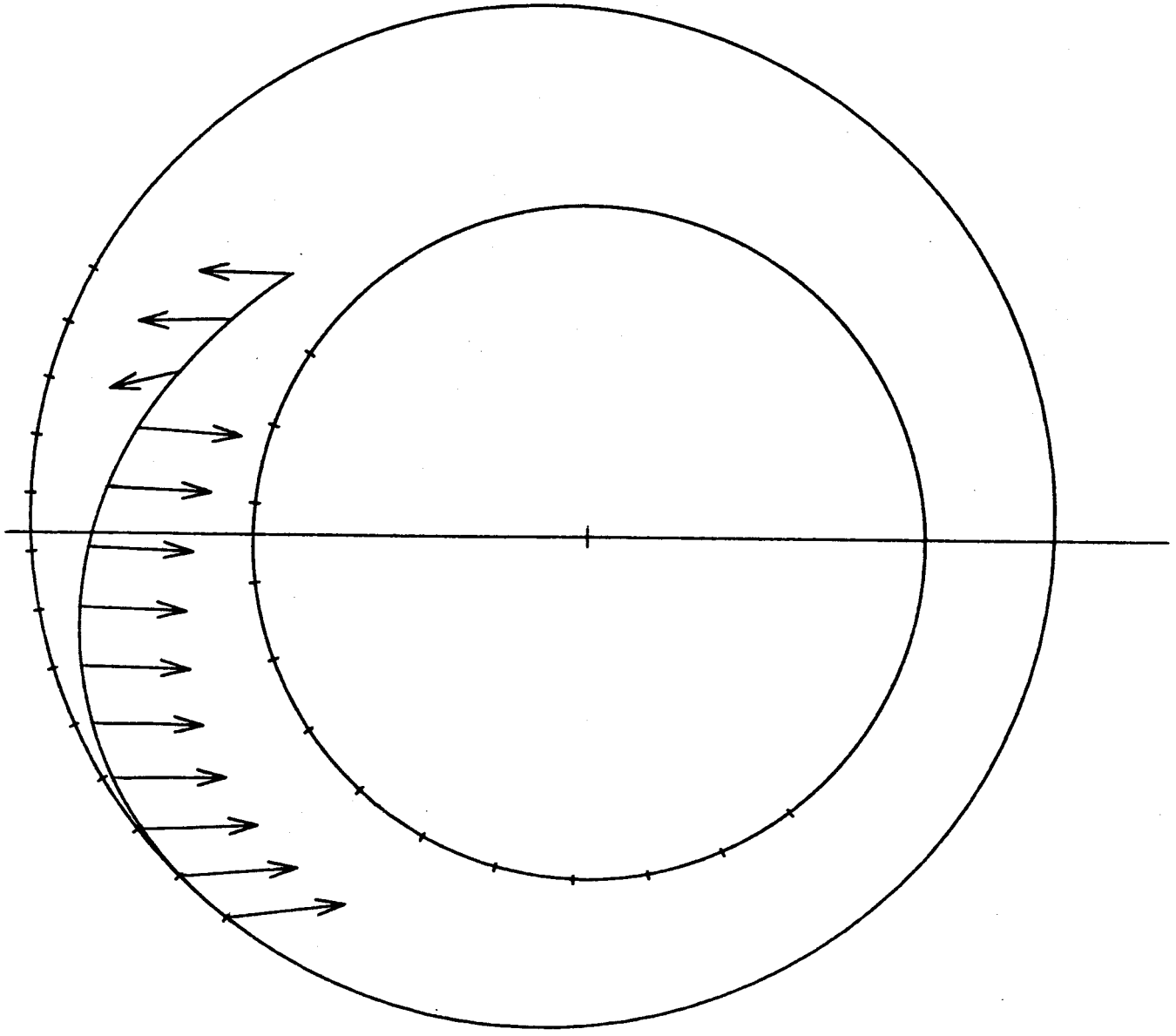
FIG. 6.3

# EARTH - MARS TRAJECTORY

## ECLIPTIC PLANE PROJECTION

## TRIP TIME = 270.9 DAYS

## 1.000 PCT ERRORS AT 106.4 DAYS



TIME INTERVAL = 13.7 DAYS

⟶ THRUST ACCELERATION

SCALE 0.5 INCH = .001 M/SEC2

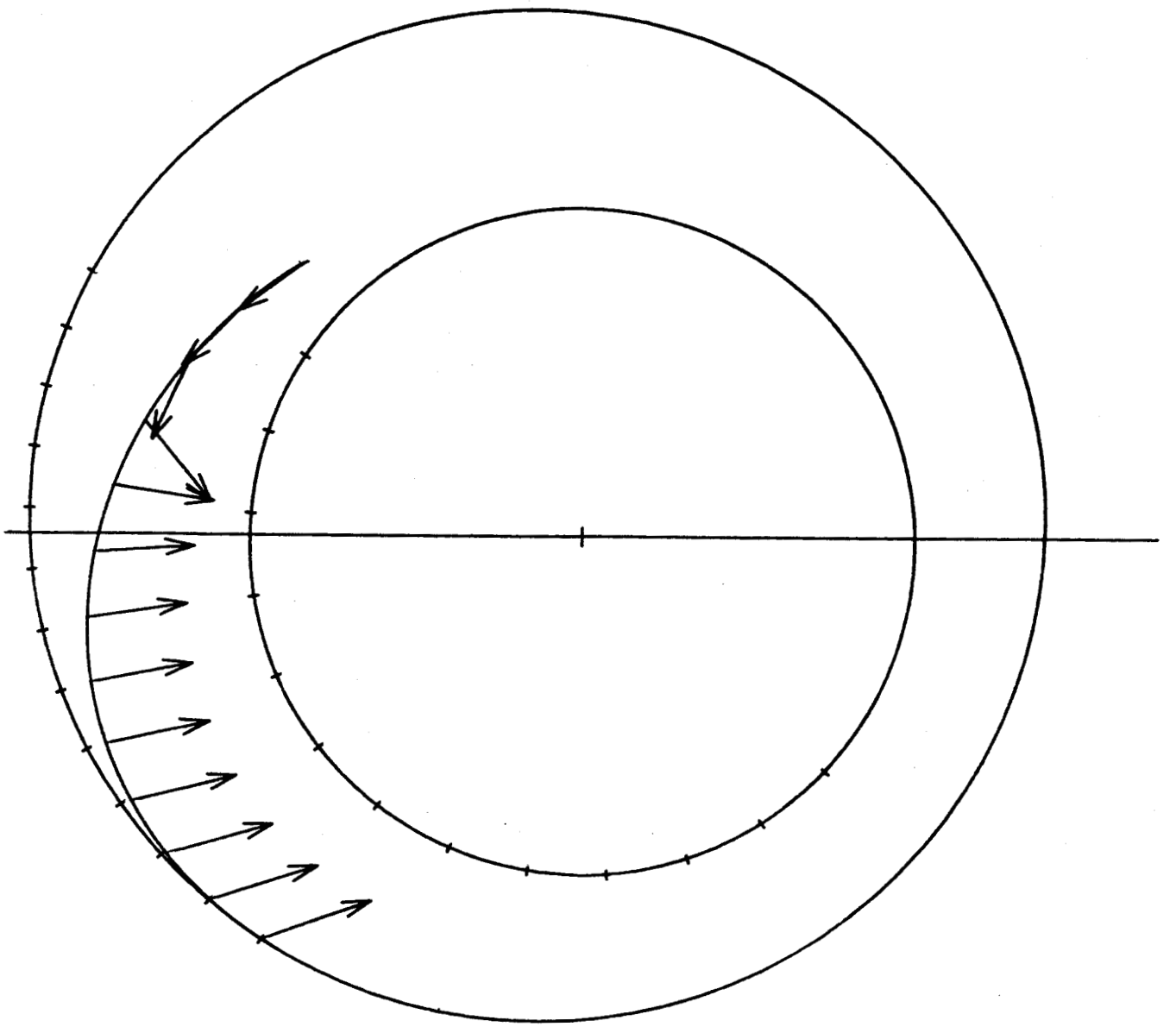
FIG 6.4

# EARTH - MARS TRAJECTORY

## ECLIPTIC PLANE PROJECTION

## TRIP TIME = 280.7 DAYS

## 2.000 PCT ERRORS AT 106.4 DAYS



TIME INTERVAL = 14.5 DAYS

⟶ THRUST ACCELERATION

SCALE 0.5 INCH = .001 M/SEC2

FIG. 6.5

# EARTH - MARS TRAJECTORY

## PRIMER VECTOR HISTORY FROM 106.4 DAYS

## 0 AND 2 PERCENT GUIDANCE ERRORS

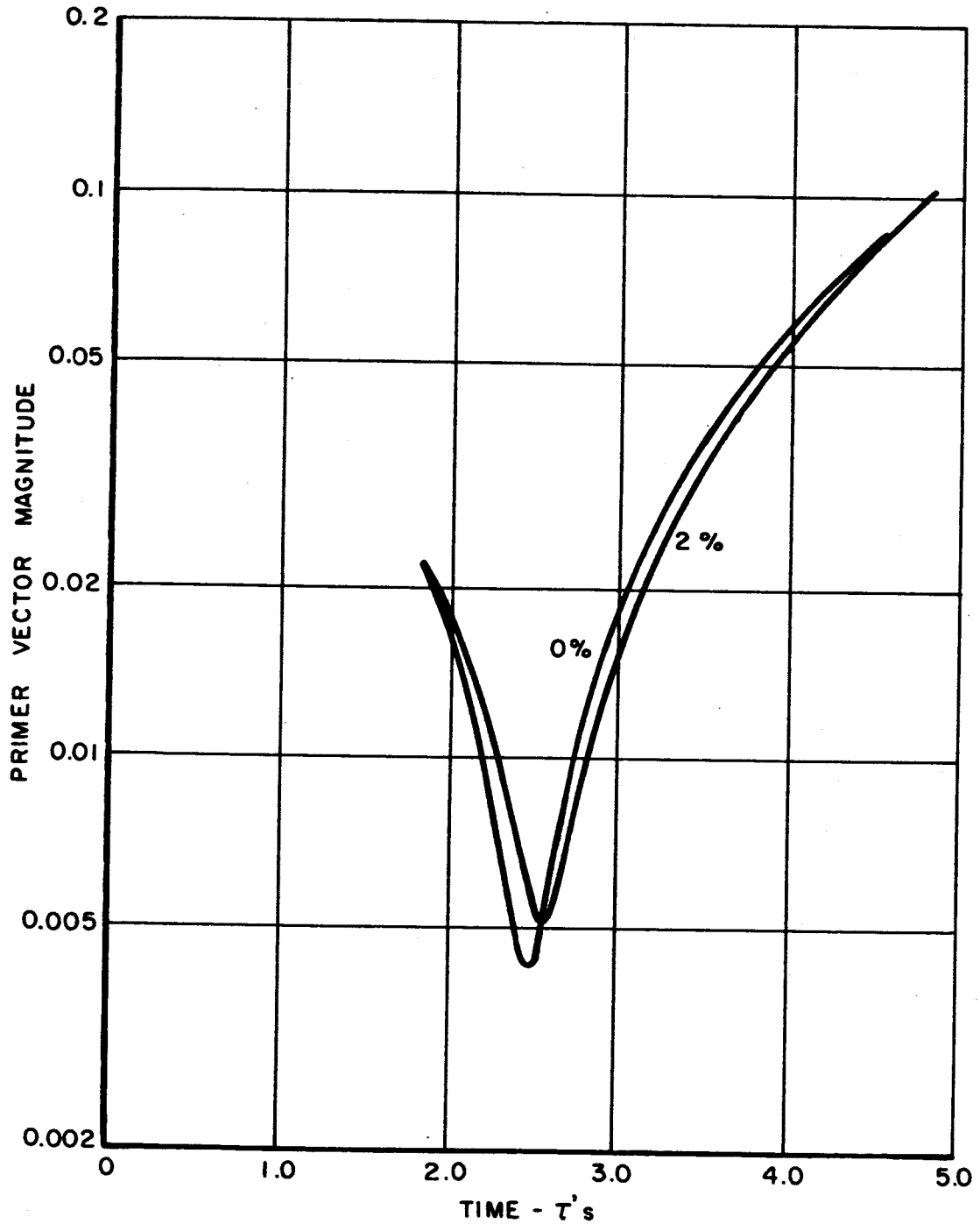


FIG. 6.6

# EARTH - MARS TRAJECTORY

## IN - PLANE CONTROL FROM 106.4 DAYS

## 0, 1, AND 2 PERCENT GUIDANCE ERRORS

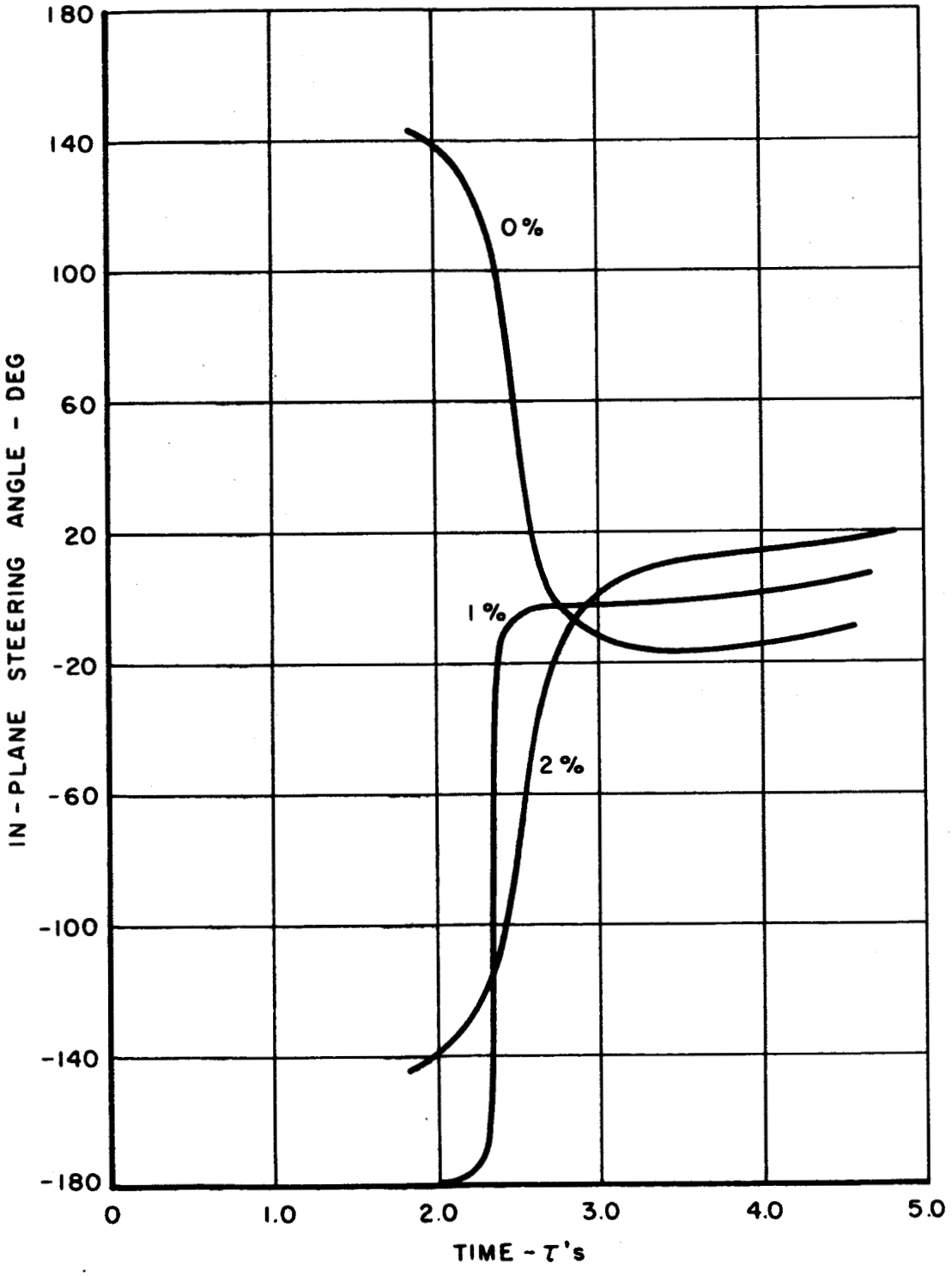


FIG. 67

# EARTH - MARS TRAJECTORY

## OUT - OF - PLANE CONTROL FROM 106.4 DAYS

## 0, 1, AND 2 PERCENT GUIDANCE ERRORS

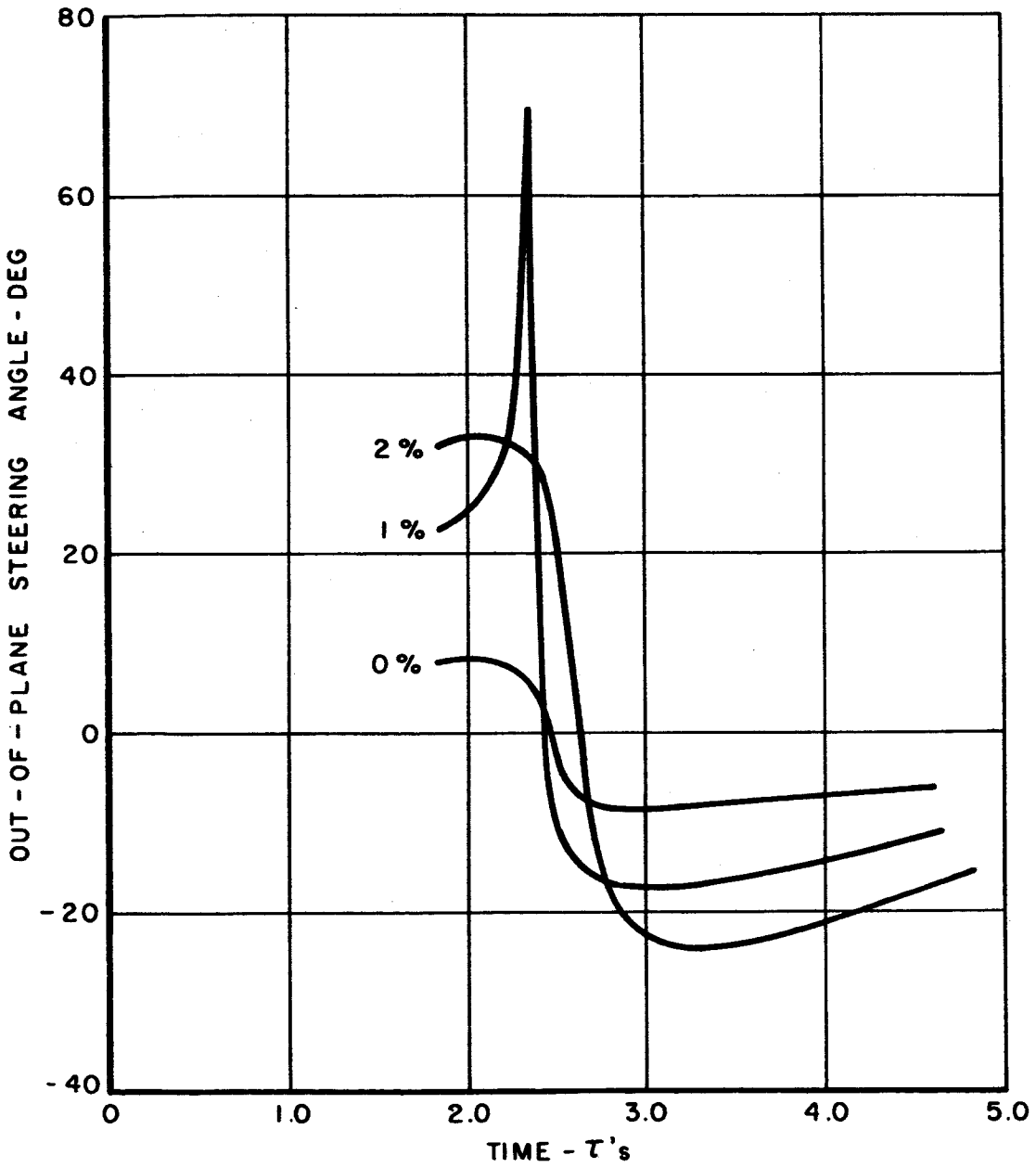


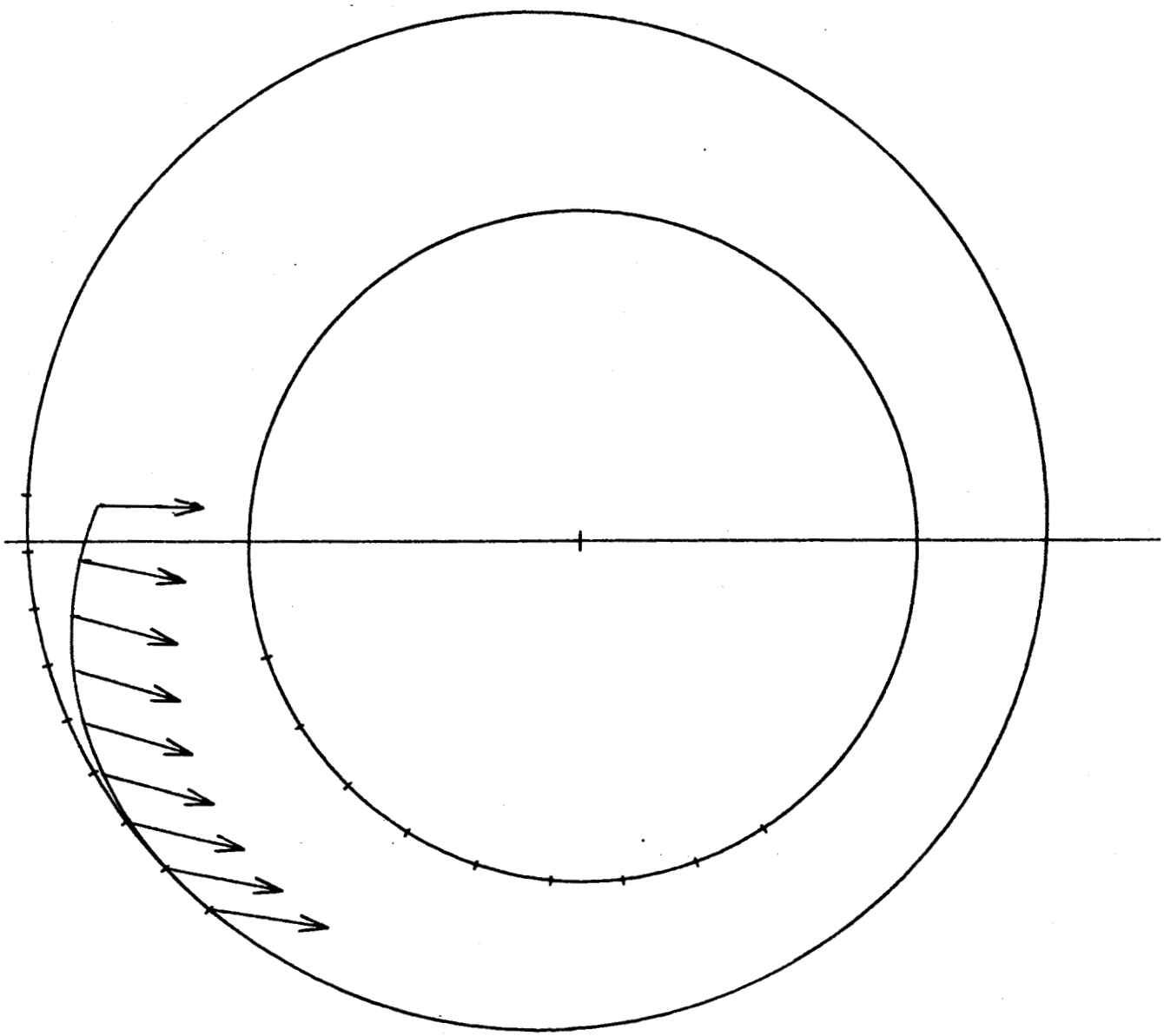
FIG. 7.0

# EARTH - MARS TRAJECTORY

## ECLIPTIC PLANE PROJECTION

## TRIP TIME = 266.1 DAYS

## 0.000 PCT ERRORS AT 159.6 DAYS



TIME INTERVAL = 13.3 DAYS

⟶ THRUST ACCELERATION

SCALE 0.5 INCH = .001 M/SEC2

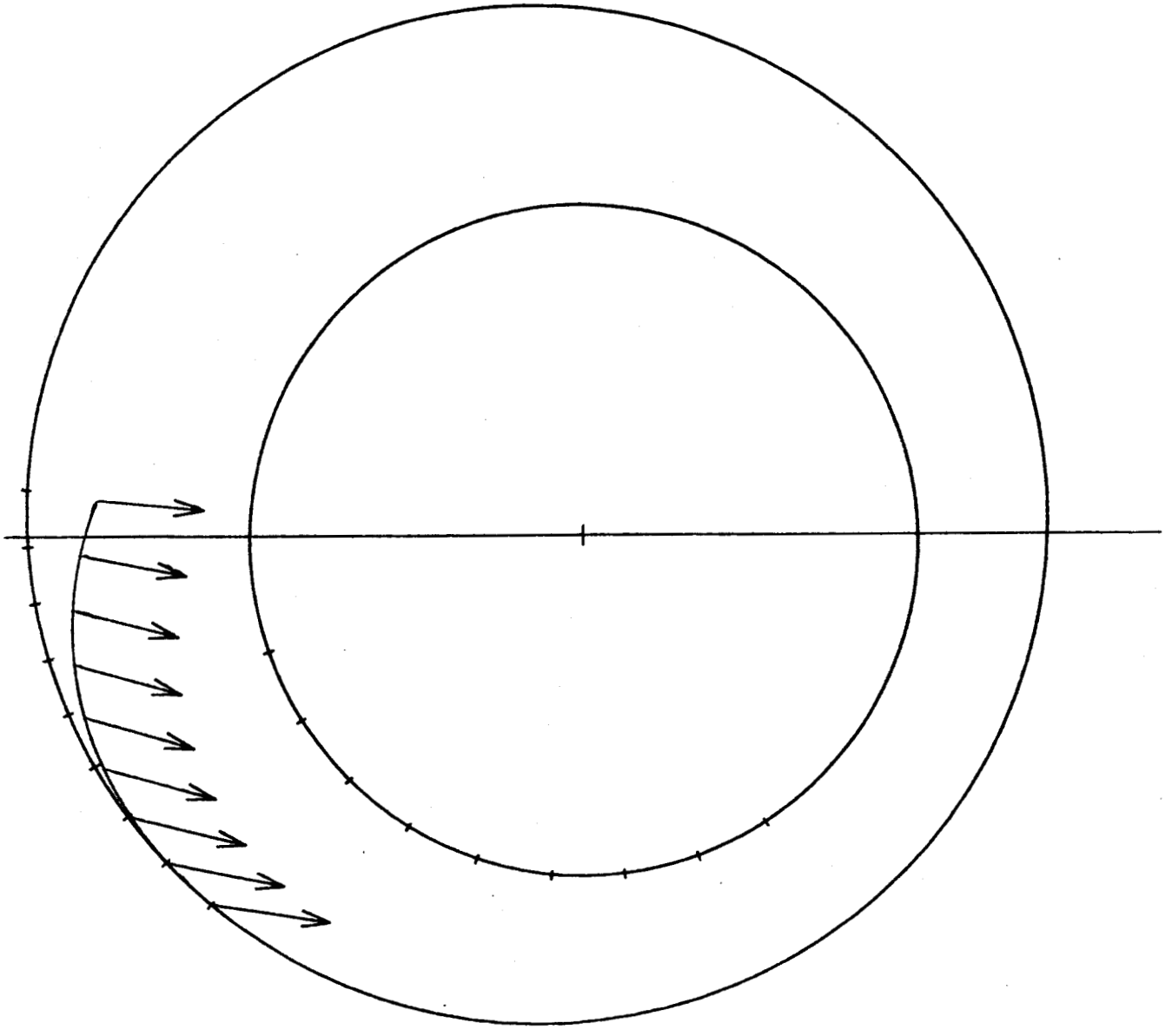
FIG. 7.1

# EARTH - MARS TRAJECTORY

## ECLIPTIC PLANE PROJECTION

## TRIP TIME = 266.1 DAYS

## 0.002 PCT ERRORS AT 159.6 DAYS



TIME INTERVAL = 13.3 DAYS

⟶ THRUST ACCELERATION

SCALE 0.5 INCH = .001 M/SEC2

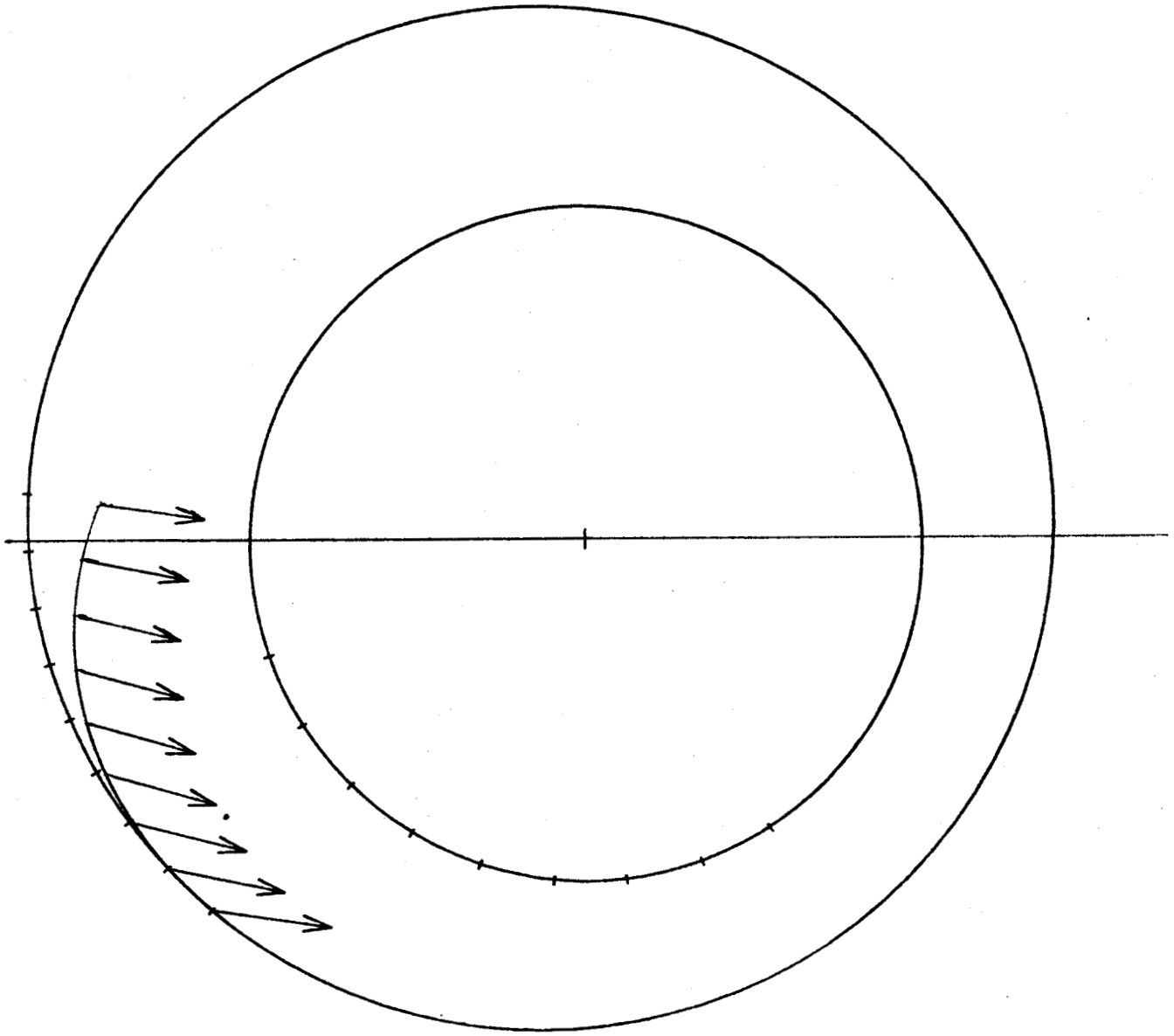
FIG. 7.2

# EARTH - MARS TRAJECTORY

## ECLIPTIC PLANE PROJECTION

## TRIP TIME = 266.1 DAYS

## 0.003 PCT ERRORS AT 159.6 DAYS



TIME INTERVAL = 13.3 DAYS

⟶ THRUST ACCELERATION

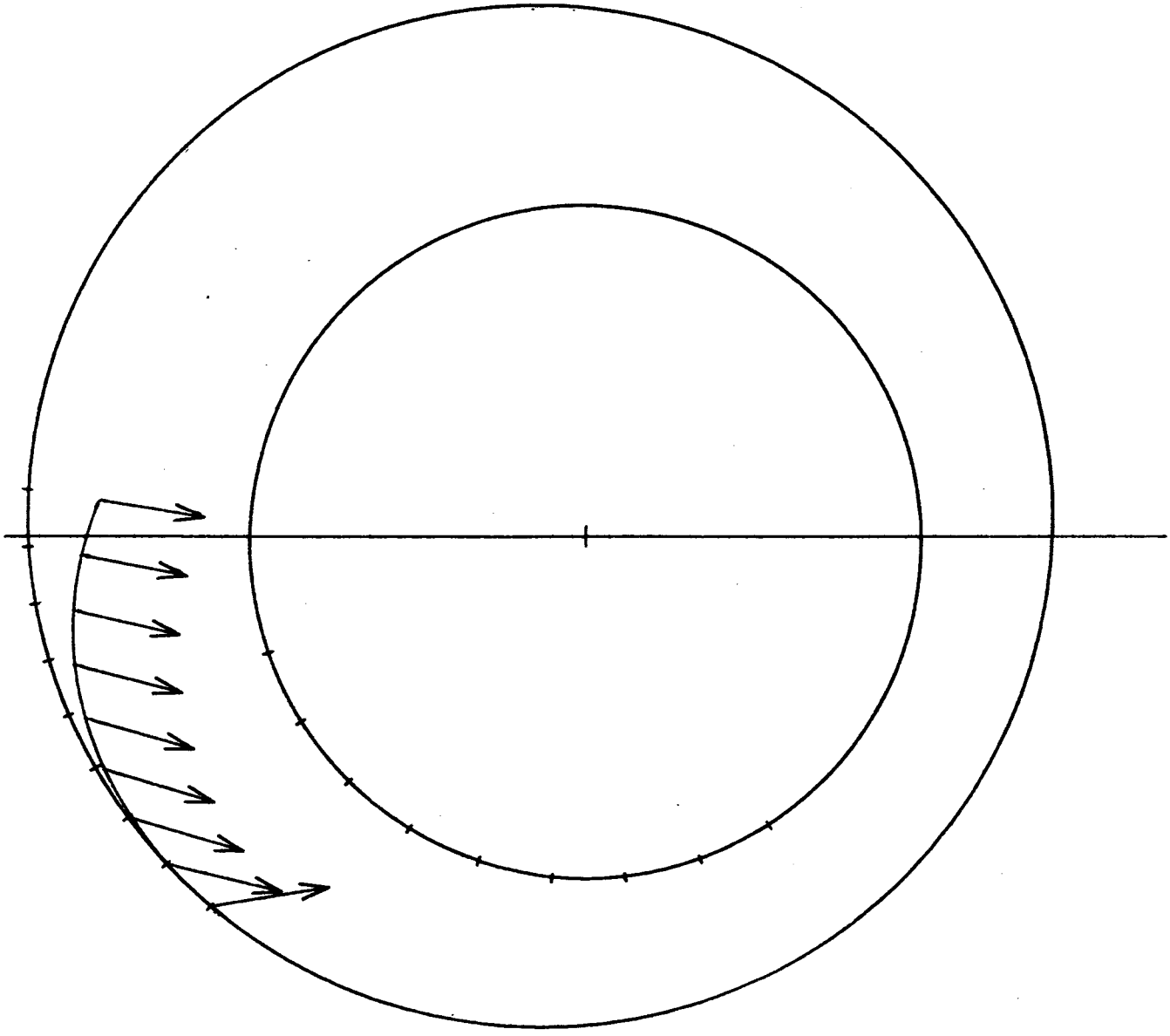
SCALE 0.5 INCH = .001 M/SEC2

# EARTH - MARS TRAJECTORY

## ECLIPTIC PLANE PROJECTION

## TRIP TIME = 266.2 DAYS

## .0042 PCT ERRORS AT 159.6 DAYS



TIME INTERVAL = 13.3 DAYS

⟶ THRUST ACCELERATION

SCALE 0.5 INCH = .001 M/SEC2

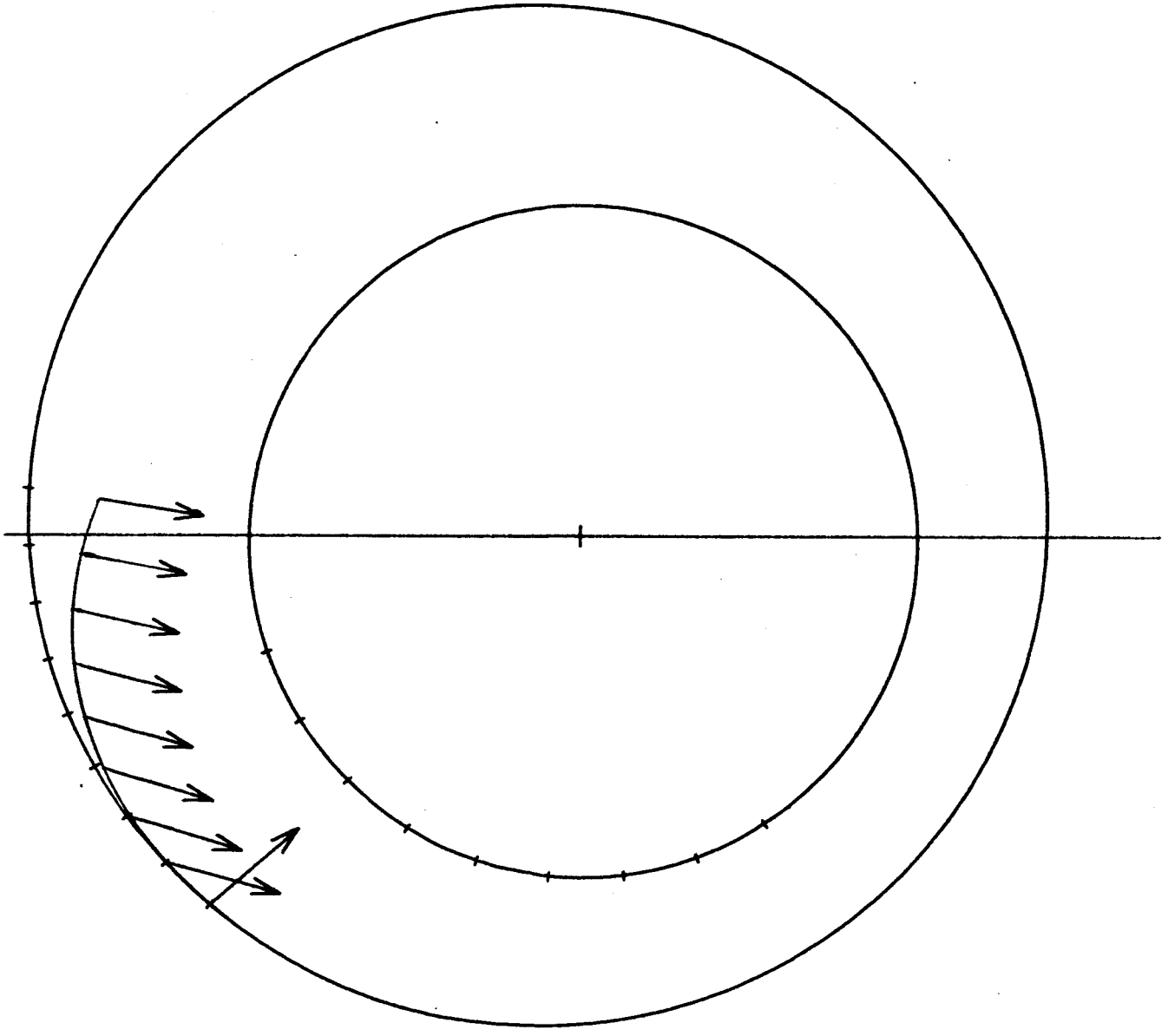
FIG. 7.4

# EARTH - MARS TRAJECTORY

## ECLIPTIC PLANE PROJECTION

## TRIP TIME = 266.4 DAYS

## .0046 PCT ERRORS AT 159.6 DAYS



TIME INTERVAL = 13.3 DAYS

⟶ THRUST ACCELERATION

SCALE 0.5 INCH = .001 M/SEC2

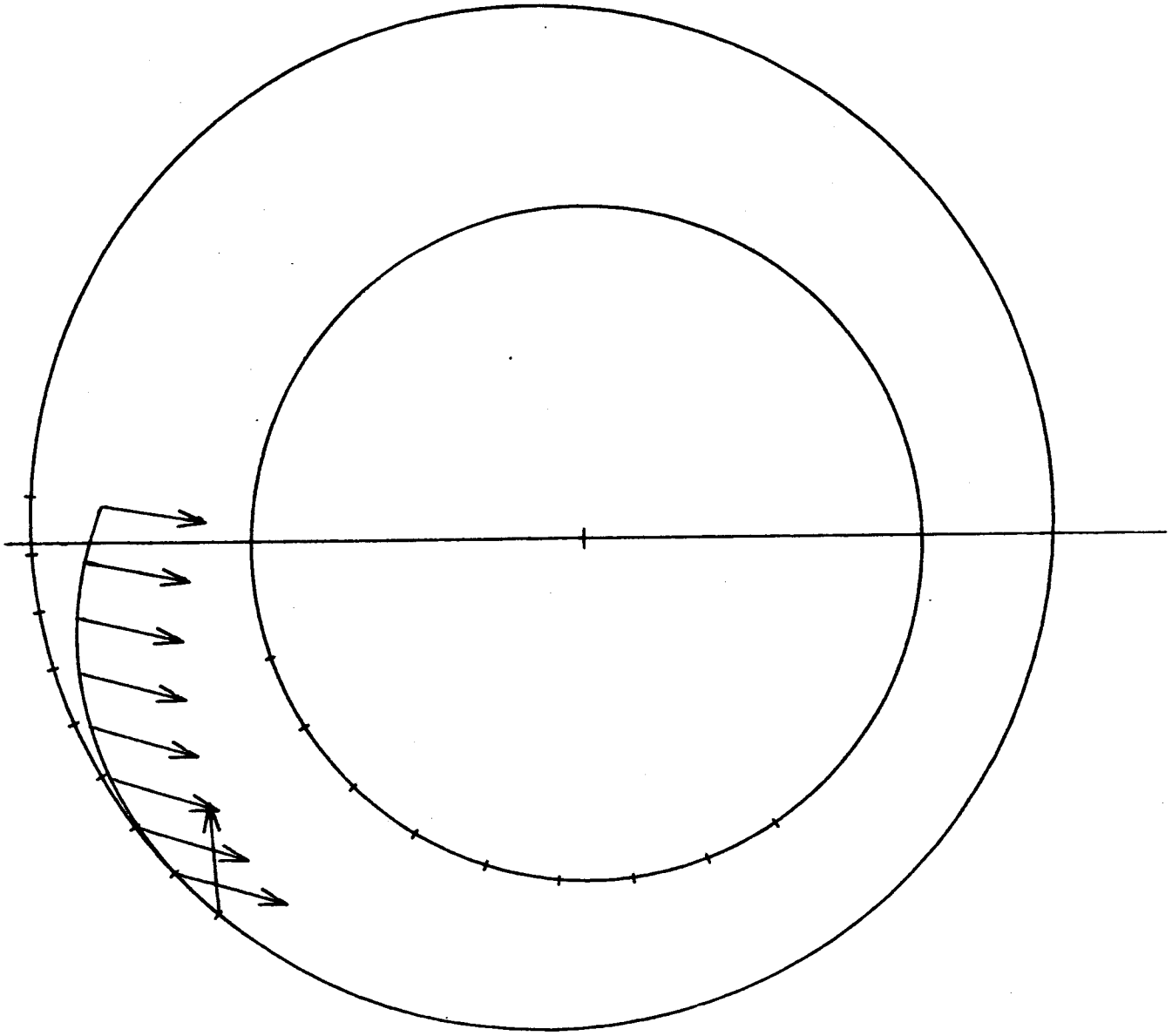
FIG. 7.5

# EARTH - MARS TRAJECTORY

## ECLIPTIC PLANE PROJECTION

## TRIP TIME = 267.0 DAYS

## .0050 PCT ERRORS AT 159.6 DAYS



TIME INTERVAL = 13.4 DAYS

⟶ THRUST ACCELERATION

SCALE 0.5 INCH = .001 M/SEC2

FIG. 7.6

# EARTH - MARS TRAJECTORY

## PRIMER VECTOR HISTORY FROM 159.6 DAYS

## 0, .002, .003, .0034 AND .005 PERCENT GUIDANCE ERRORS

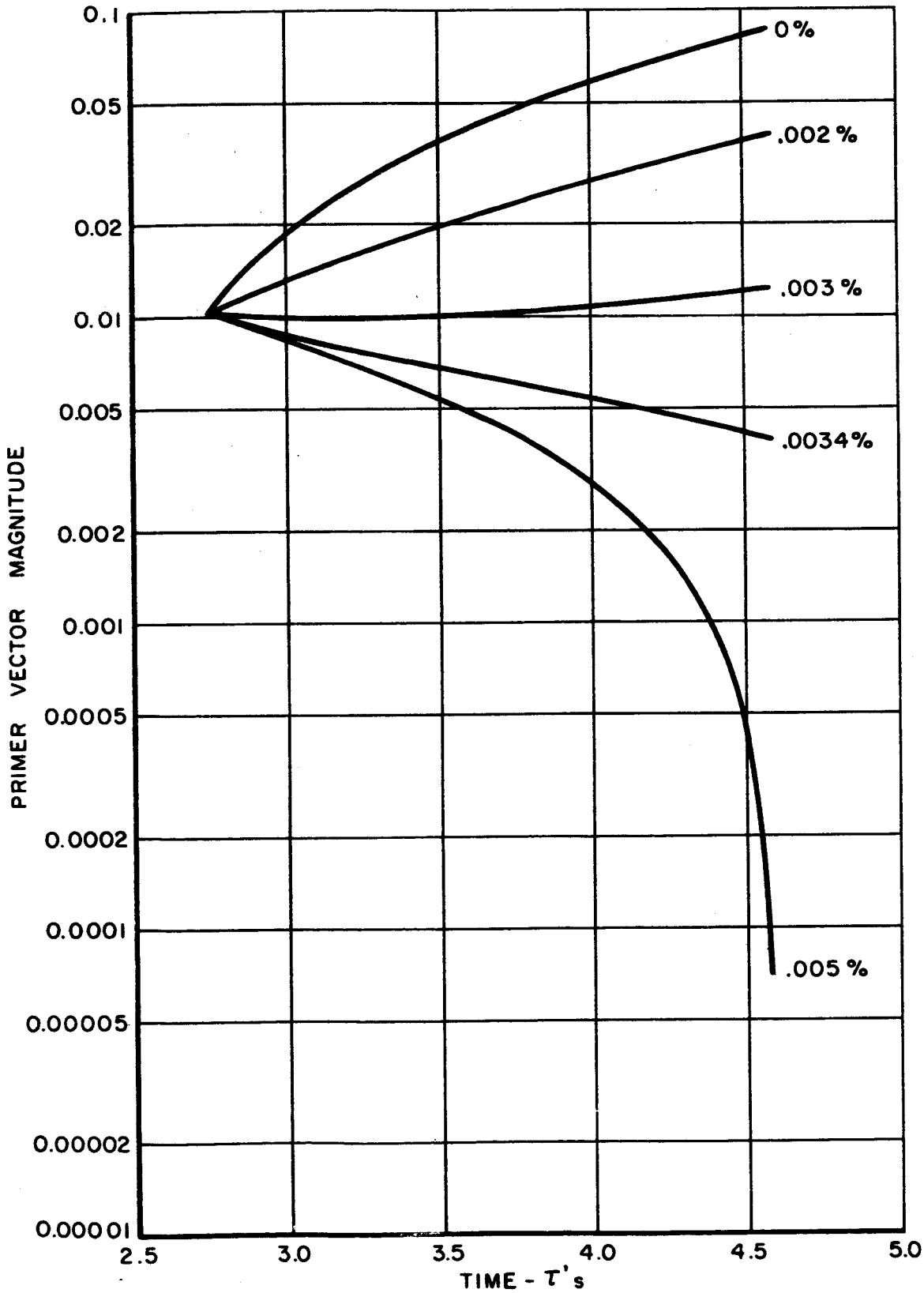


FIG. 7.7

# EARTH - MARS TRAJECTORY

## IN - PLANE CONTROL FROM 159.6 DAYS

## 0 AND .005 PERCENT GUIDANCE ERRORS

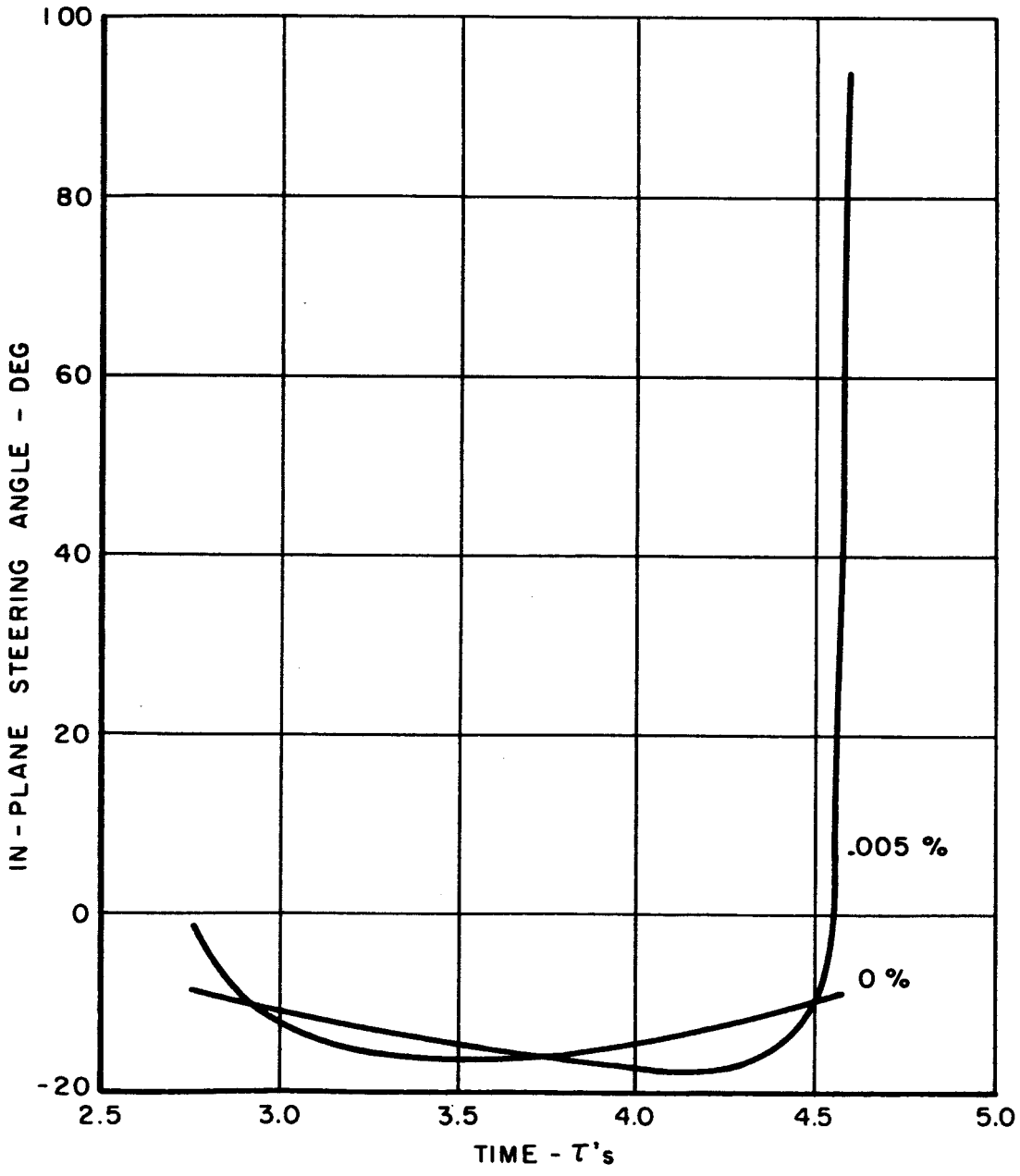
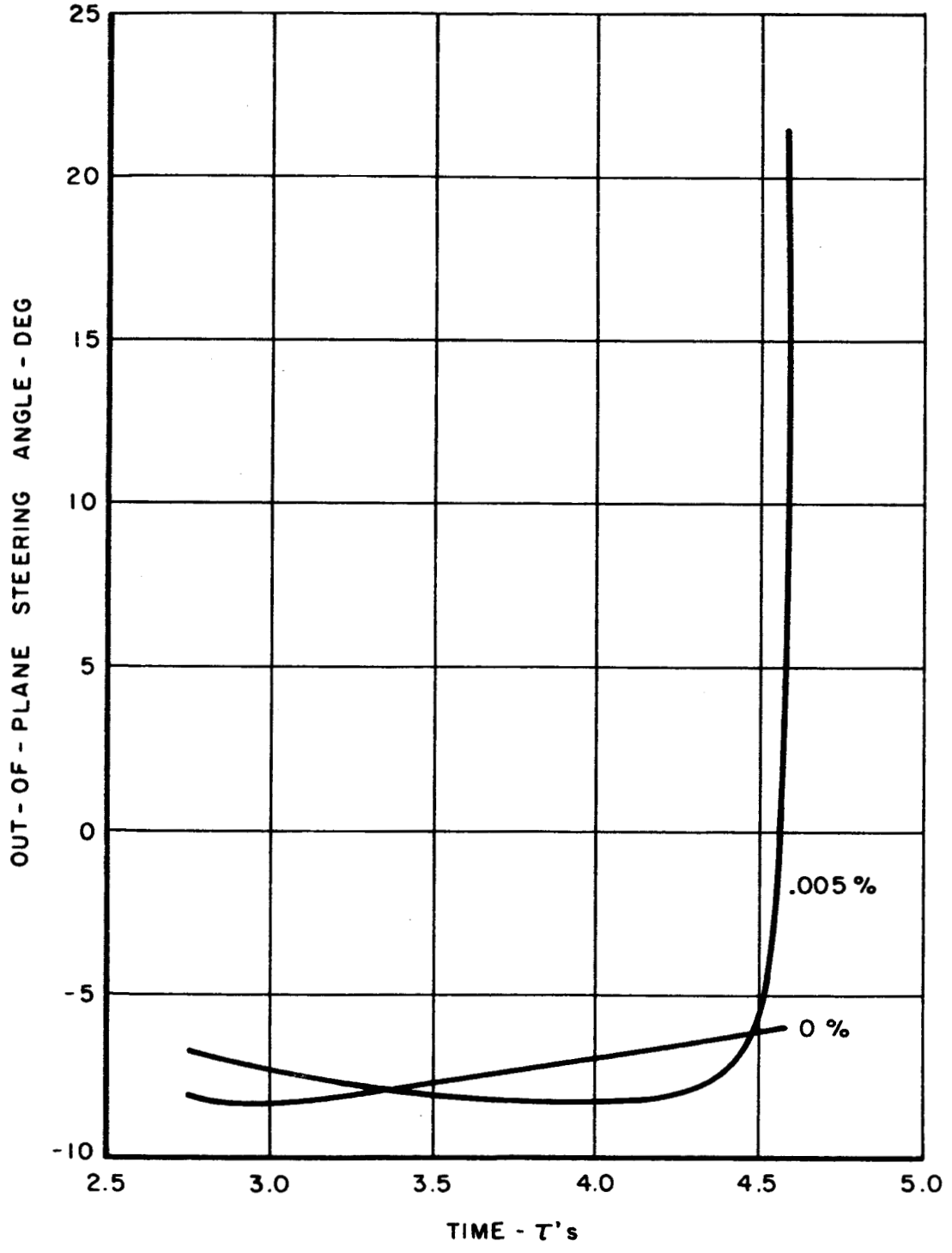


FIG. 7.8

# EARTH - MARS TRAJECTORY

## OUT - OF - PLANE CONTROL FROM 159.6 DAYS

## 0 AND .005 PERCENT GUIDANCE ERRORS

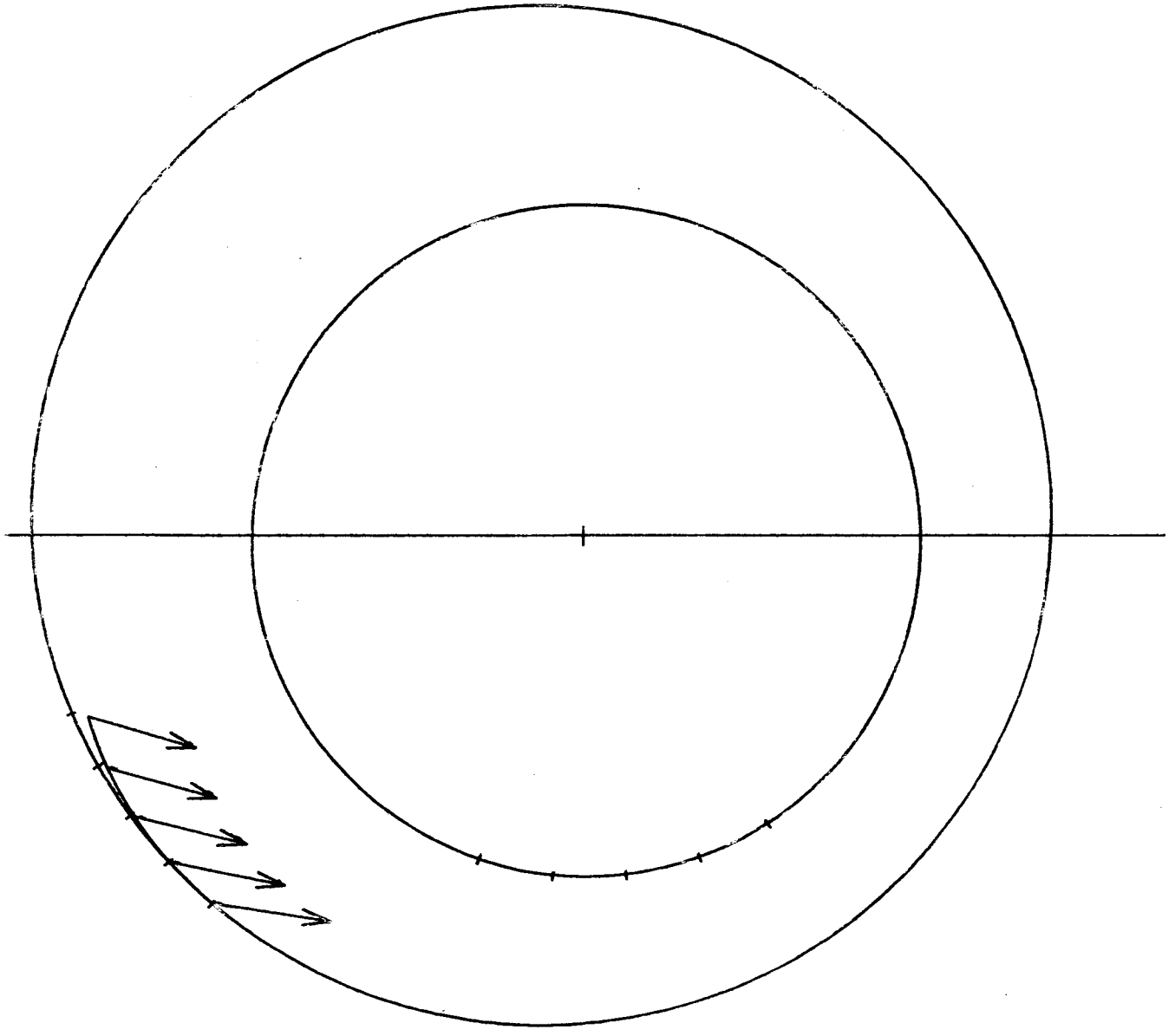


# EARTH - MARS TRAJECTORY

## ECLIPTIC PLANE PROJECTION

## TRIP TIME = 266.1 DAYS

## 0.000 PCT ERRORS AT 212.9 DAYS



TIME INTERVAL = 13.3 DAYS

⟶ THRUST ACCELERATION

SCALE 0.5 INCH = .001 M/SEC2

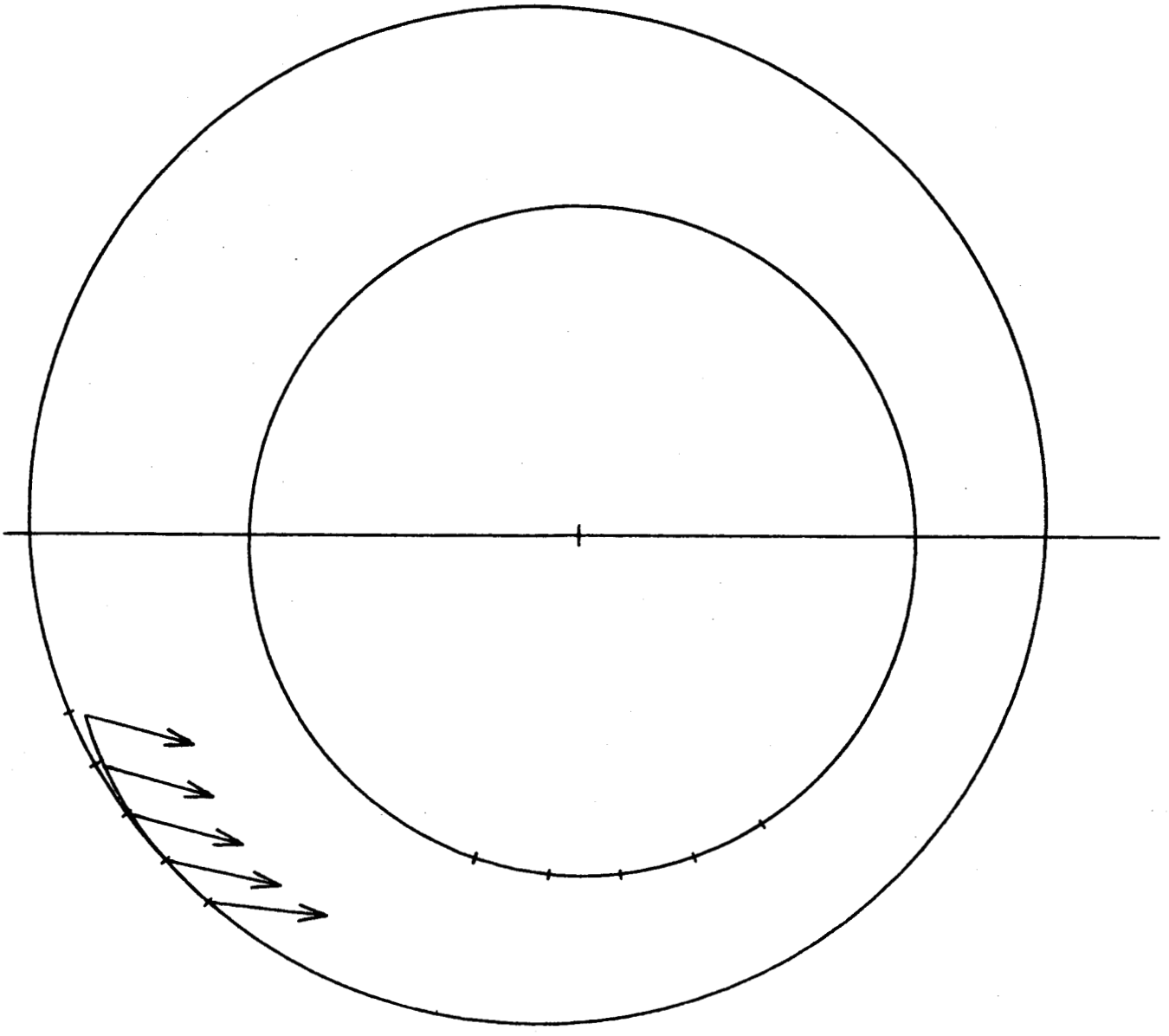
FIG. 6.1

# EARTH - MARS TRAJECTORY

## ECLIPTIC PLANE PROJECTION

## TRIP TIME = 266.1 DAYS

## .0002 PCT ERRORS AT 212.9 DAYS



TIME INTERVAL = 13.3 DAYS

⟶ THRUST ACCELERATION

SCALE 0.5 INCH = .001 M/SEC2

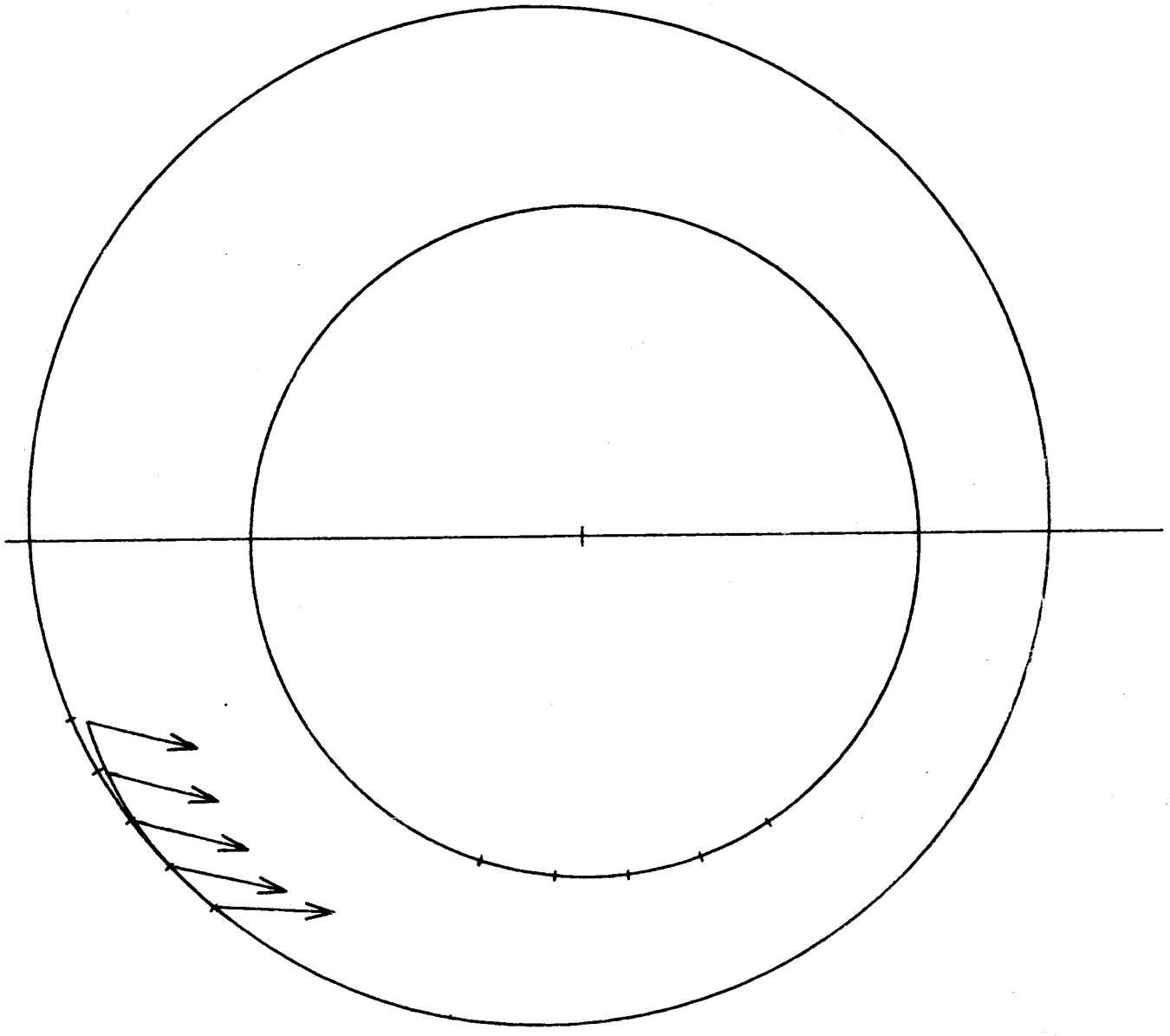
FIG. 5.2

# EARTH - MARS TRAJECTORY

## ECLIPTIC PLANE PROJECTION

## TRIP TIME = 266.1 DAYS

## .0003 PCT ERRORS AT 212.9 DAYS



TIME INTERVAL = 13.3 DAYS

⟶ THRUST ACCELERATION

SCALE 0.5 INCH = .001 M/SEC2

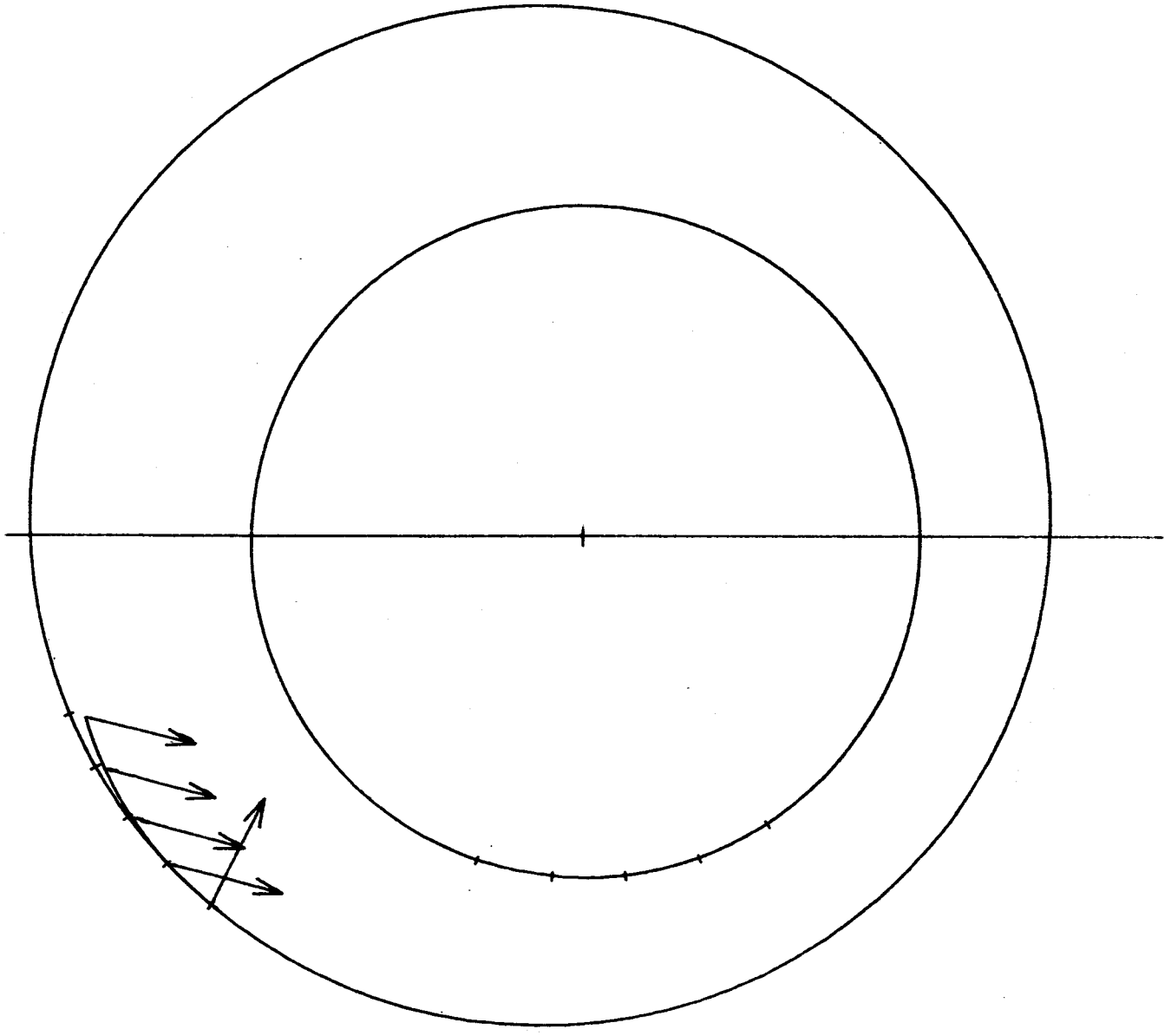
FIG. 8.3

# EARTH - MARS TRAJECTORY

## ECLIPTIC PLANE PROJECTION

## TRIP TIME = 266.3 DAYS

## .0005 PCT ERRORS AT 212.9 DAYS



TIME INTERVAL = 13.3 DAYS

⟶ THRUST ACCELERATION

SCALE 0.5 INCH = .001 M/SEC2

FIG. 8.4

# EARTH - MARS TRAJECTORY

## PRIMER VECTOR HISTORY FROM 212.8 DAYS
## 0, .0001, .0002, .0003 AND .0005 PERCENT GUIDANCE ERRORS

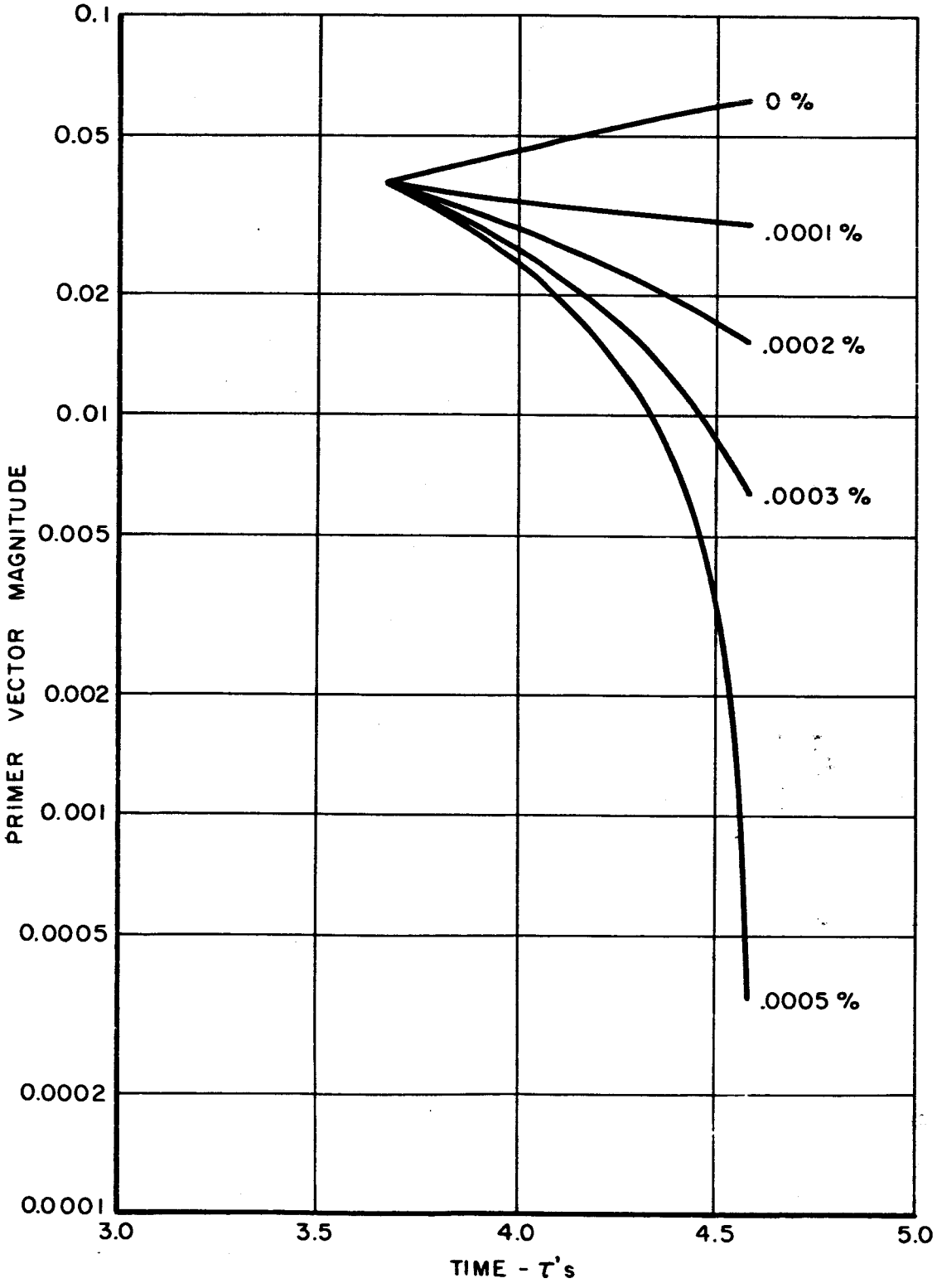


FIG. 8.5

# EARTH - MARS TRAJECTORY

## IN-PLANE CONTROL FROM 212.8 DAYS

## 0 AND .0005 PERCENT GUIDANCE ERRORS

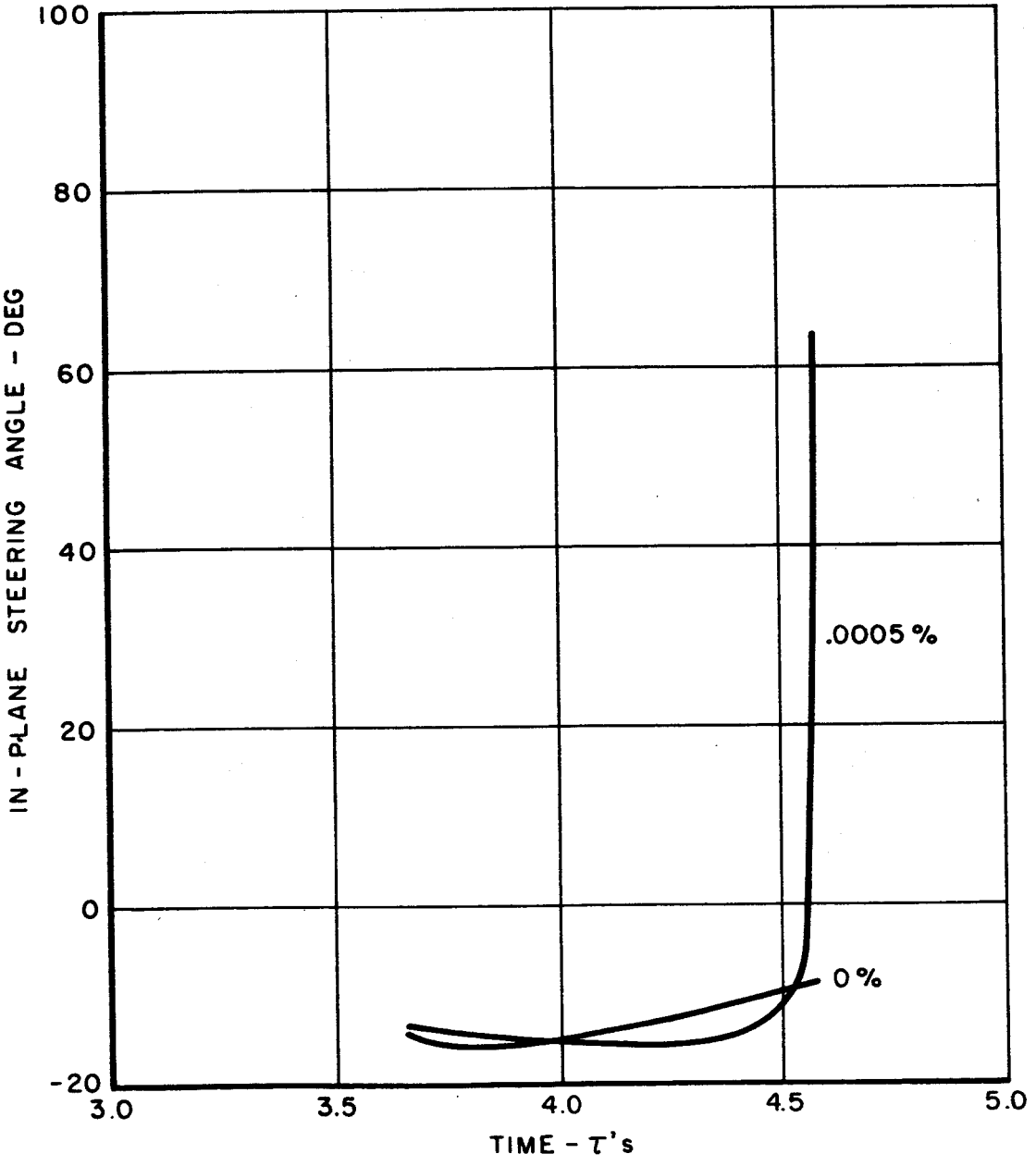


FIG. 8.6

# EARTH - MARS TRAJECTORY

## OUT - OF - PLANE CONTROL FROM 212.8 DAYS

## 0 AND .0005 PERCENT GUIDANCE ERRORS

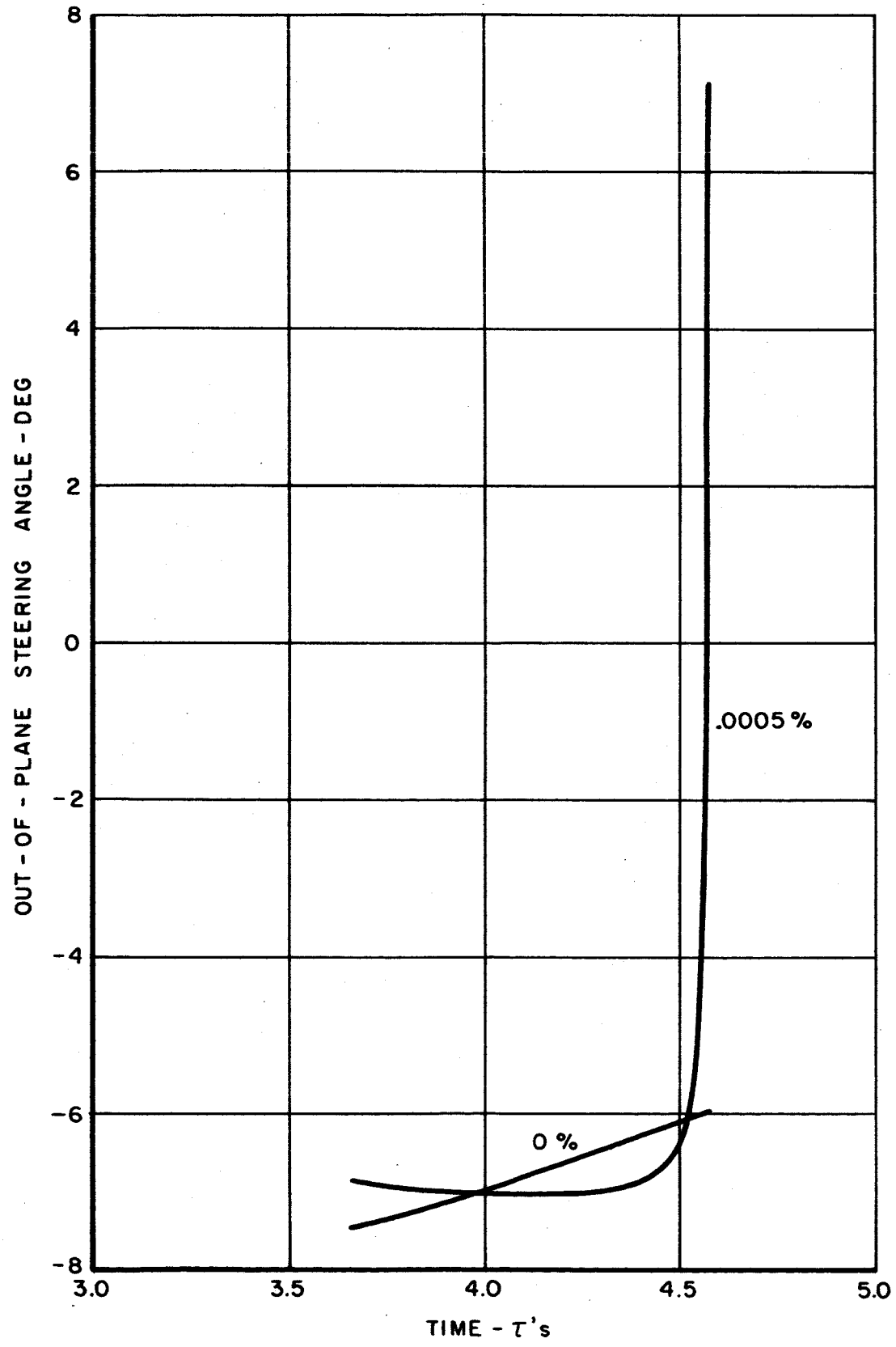


FIG. 9.0

# EARTH - MARS TRAJECTORY

## CHANGE OF IN-PLANE CONTROL FOR RADIAL GUIDANCE ERRORS AT 200 DAYS

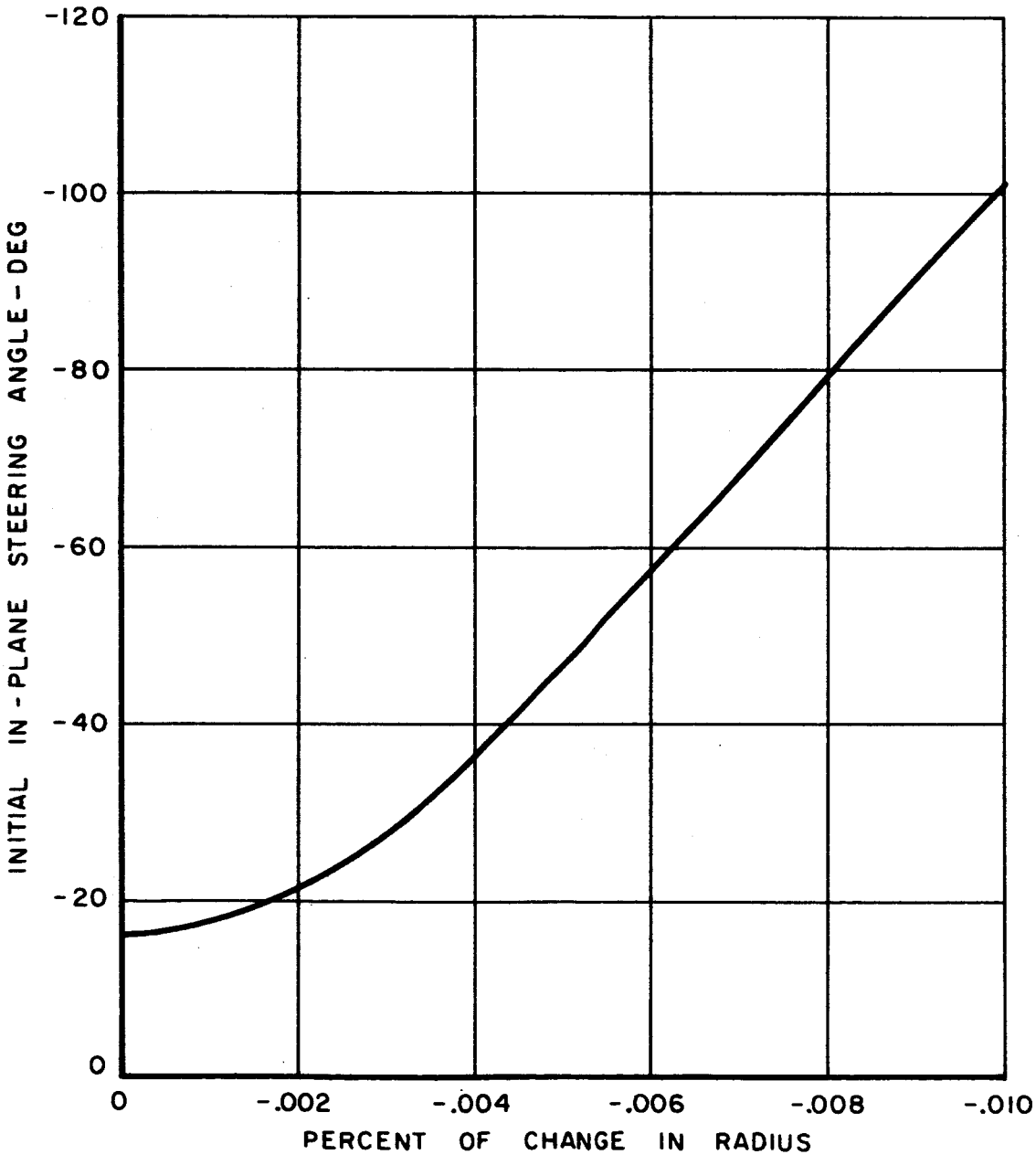


FIG. 9.1

# EARTH - MARS TRAJECTORY

## IN - PLANE CONTROL FROM 200 DAYS

## 0, .006, AND .010 PERCENT GUIDANCE ERRORS IN RADIUS ONLY

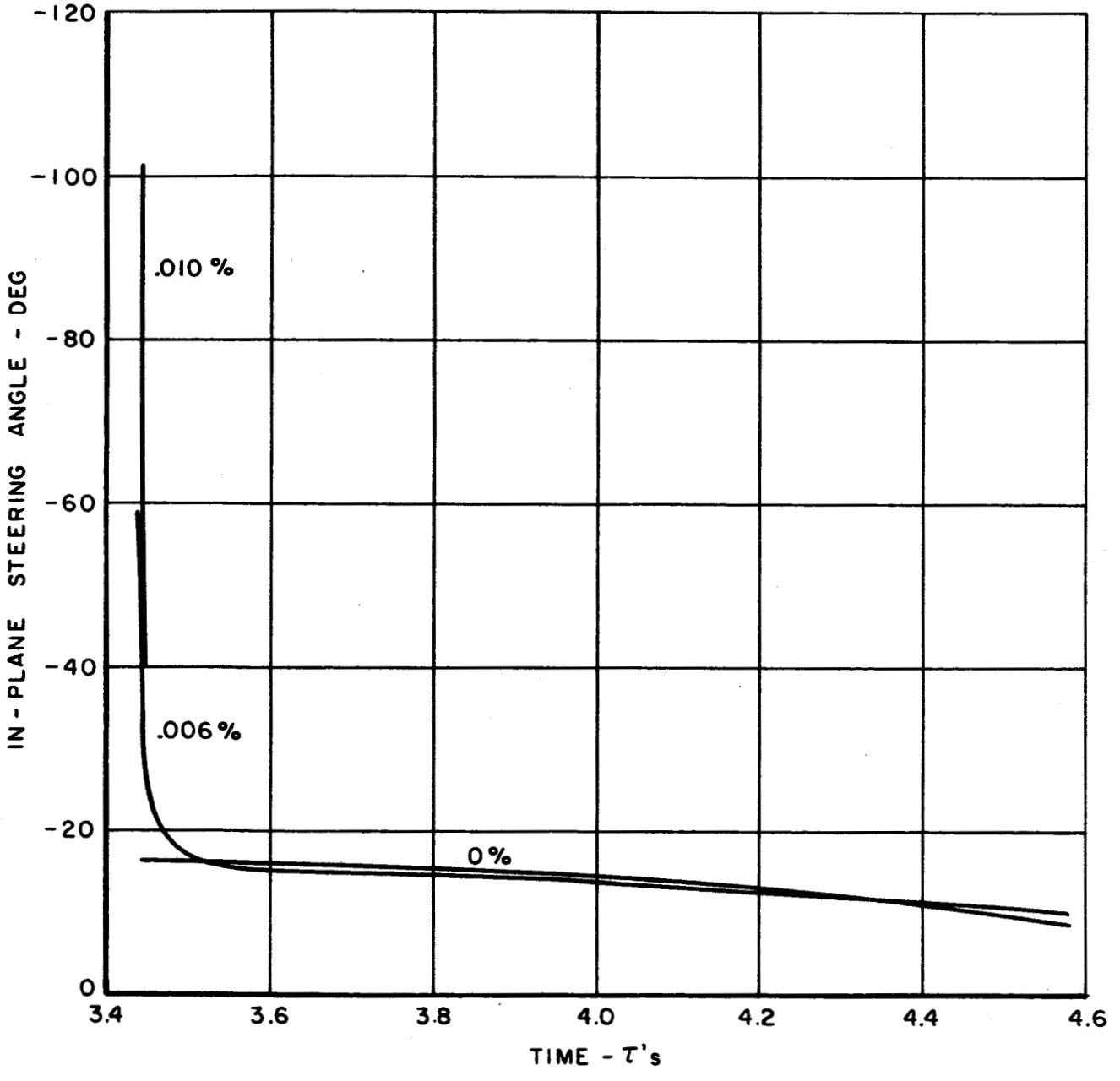


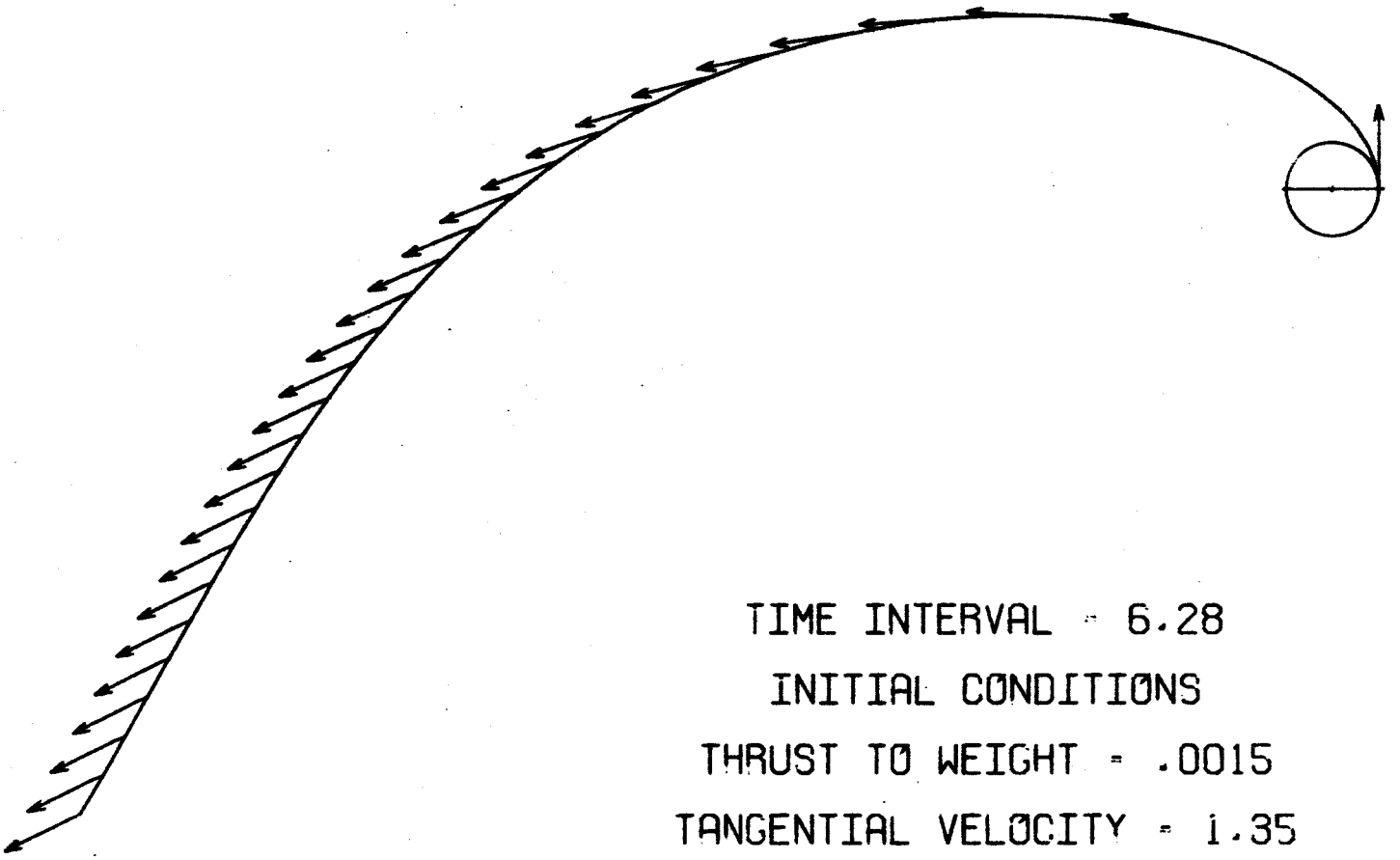
FIG. 10.0

# ESCAPE TYPE TRAJECTORY

## NON-DIMENSIONAL UNITS

## RADIUS RATIO = 1/30

## TIME = 176.3



TIME INTERVAL = 6.28

INITIAL CONDITIONS

THRUST TO WEIGHT = .0015

TANGENTIAL VELOCITY = 1.35

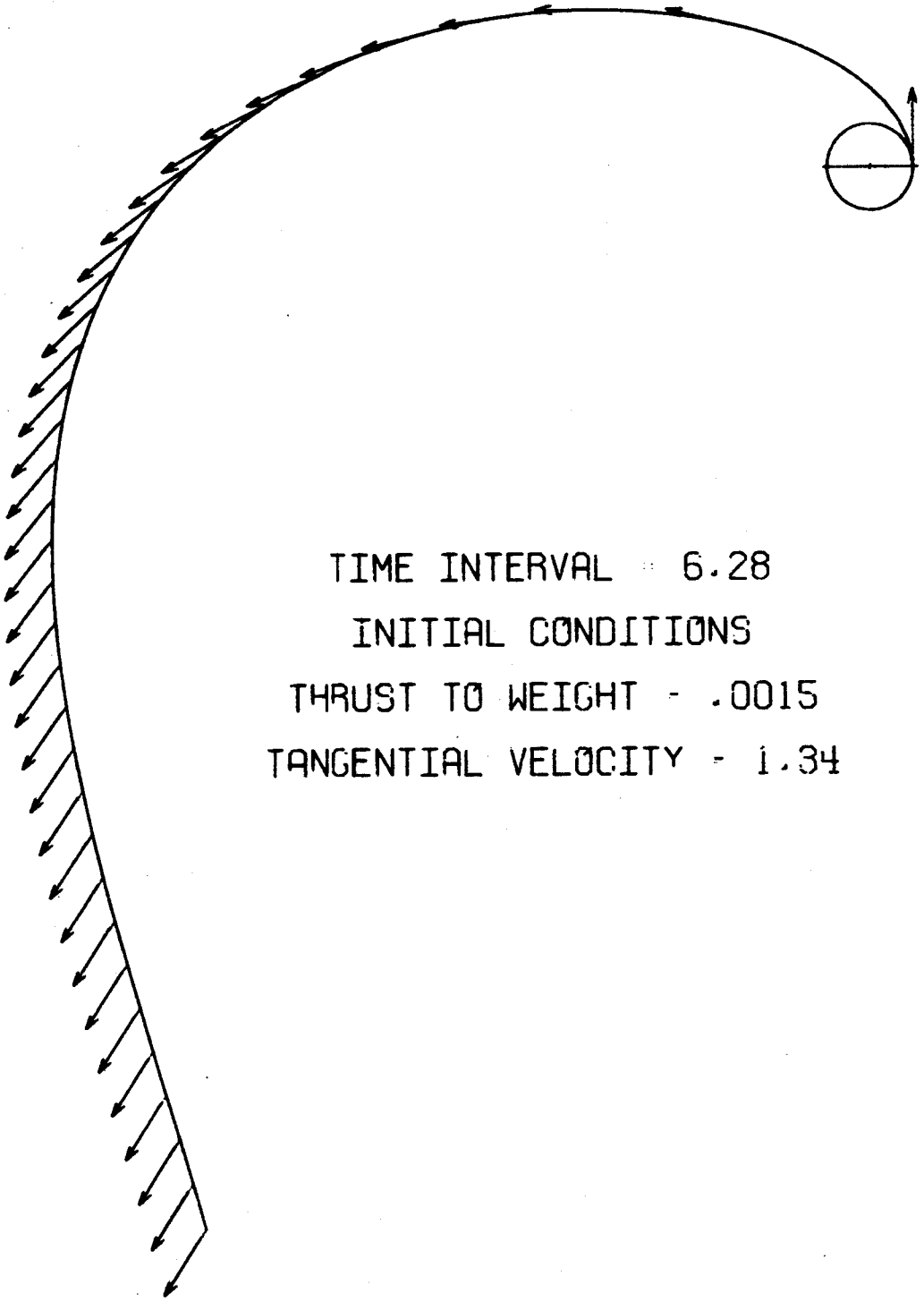
FIG. 10.1

# ESCAPE TYPE TRAJECTORY

## NON-DIMENSIONAL UNITS

## RADIUS RATIO = 1/29

## TIME = 224.9



TIME INTERVAL = 6.28

INITIAL CONDITIONS

THRUST TO WEIGHT = .0015

TANGENTIAL VELOCITY = 1.34